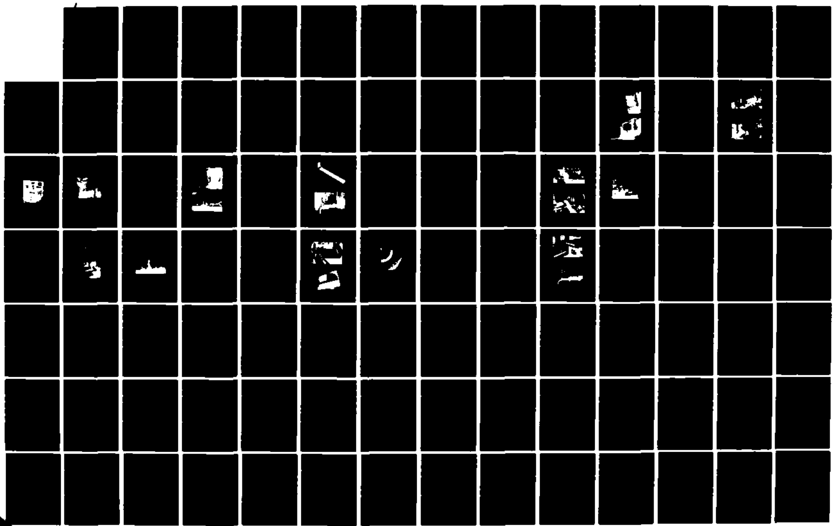
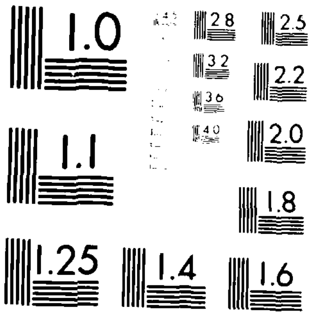
AD-A152 224 DESIGN AND CONSTRUCTION OF A COMPUTER CONTROLLED MIROTHERMOCOUPLE PROBE FOR THE STUDY OF BUOYANT JETS (U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA 1/2

UNCLASSIFIED R J MATOUSHEK SEP 84 F/G 9/2 NL





MICROCOPY RESOLUTION TEST CHART

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

DESIGN AND CONSTRUCTION OF A COMPUTER
CONTROLLED MICROTHERMOCOUPLE PROBE
FOR THE STUDY OF BUOYANT JETS

by

Ronald John Matoushek

September 1984

Thesis Advisor: William G. Culbreth

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A152224	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Design and Construction of a Computer Controlled Microthermocouple Probe for the Study of Buoyant Jets		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis September 1984
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Ronald John Matoushek		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		12. REPORT DATE September 1984
		13. NUMBER OF PAGES 110
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Buoyant Jet, Computer-aided Data Acquisition, Microthermocouple

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A computer-aided data acquisition system was developed and a microthermocouple probe constructed to obtain thermal distributions in turbulent buoyant jets exposed to a cross-flowing ambient fluid. The system performed high speed temperature measurements as a microthermocouple probe was automatically traversed through a sequence of preprogrammed positions under the control of a microcomputer. Operability of the apparatus

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

#20 - ABSTRACT - (CONTINUED)

was demonstrated by measuring temperature distributions in planes perpendicular to the streamwise axis of jets from which contour plots of temperature were generated. Using temperature distributions along with velocity distributions allow buoyant jet characteristics to be computed, including the entrainment rate of ambient fluid, jet trajectory, and heat transfer to the ambient. The experimental technique is discussed and temperature contour plots for a jet at various planes are presented.

[illegible]

S/N 0102-LF-014-6601

Approved for public release; distribution unlimited.

Design and Construction of a Computer Controlled Microthermocouple Probe for the Study of Buoyant Jets

by

Ronald John Matoushek
Lieutenant Commander, United States Navy
B.S.E.E., Purdue University, 1974

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

September 1984

Author: Ronald John Matoushek
Ronald John Matoushek

Approved by: William G. Culbreth
William G. Culbreth, Thesis Advisor

P. J. Marto
Paul J. Marto, Chairman,
Department of Mechanical Engineering

J. M. Dyer
John Dyer, Dean of Science and Engineering

ABSTRACT

A computer-aided data acquisition system was developed and a microthermocouple probe constructed to obtain thermal distributions in turbulent buoyant jets exposed to a cross-flowing ambient fluid. The system performed high speed temperature measurements as a microthermocouple probe was automatically traversed through a sequence of preprogrammed positions under the control of a microcomputer. Operability of the apparatus was demonstrated by measuring temperature distributions in planes perpendicular to the streamwise axis of jets from which contour plots of temperature were generated. Using temperature distributions along with velocity distributions allow buoyant jet characteristics to be computed, including the entrainment rate of ambient fluid, jet trajectory, and heat transfer to the ambient. The experimental technique is discussed and temperature contour plots for a jet at various planes are presented.

TABLE OF CONTENTS

LIST OF FIGURES

NOMENCLATURE

A_{ij}	Incremental Cross-Sectional Area in the Temperature Matrix
B	Jet Half-width
b	Normalized Jet Half-width
c_p	Specific Heat
D	Diameter of the Jet at the Nozzle
D_{AB}	Binary Mass Diffusion Coefficient
F	Densiometric Froude Number
g	Acceleration of Gravity
$\dot{Q}$	Heat Transfer Rate from the Jet to the Ambient Fluid
R	Ambient-to-Nozzle Flow Ratio
R_a	Length of the Probe Arm
r	Radial Distance from the Center of the Jet
r_p	Length of the Probe
S	Schmidt Number
s	Streamwise Coordinate Along the Jet Centerline
T	Normalized Jet Temperature
T_a	Ambient Fluid Temperature
T_{ij}	Jet Temperature Within the Temperature Matrix
T_m	Centerline Jet Temperature
T_n	Nozzle Temperature
T_p	Jet Temperature As Measured by the Probe
T(r)	Temperature Within the Jet at a Radial Distance (r) from its Center

U_m	Centerline Velocity of the Jet at the Nozzle
U_o	Discharge Velocity of the Jet
$U(r)$	Velocity within the Jet at a Radial Distance "r" from its Center
U_{ij}	Jet Velocities Corresponding to Locations within the Temperature Matrix
u	Normalized Centerline Velocity
α	Entrainment Coefficient; Offset Angle of the Probe Arm
β	Offset Angle of the Probe Mounting Bracket
γ	Probe Angle of Deflection from Horizontal
θ	Local Angle of Inclination from Horizontal of the Jet Streamwise Axis
λ	Spreading Ratio
ν	Kinematic Viscosity
ρ	Density of the Jet Fluid
ρ_a	Density of the Ambient Fluid
ρ_e	Density of the Ambient Fluid at the Nozzle Exit
ρ_m	Density of the Jet at Centerline
ρ_{mo}	Centerline Density of the Jet at the Nozzle
ϕ	Angle of Inclination of the Data Plane from Horizontal

I. INTRODUCTION

Buoyant jets are very common in nature. We see them in the form of exhaust gases emitted from smoke stacks of refineries, mills and ships. We see them in the form of heated waste water expelled into the sea from power plants and from the main propulsion condensers in steam driven ships and submarines. It is no wonder that the fluid mechanics and heat transfer characteristics of buoyant jets have been of interest to environmental, civil and mechanical engineers for decades. To evaluate their ecological impact, and of most recent interest, to harness buoyant jets as a means of detecting military targets and guiding weapons, it is necessary to develop models which accurately predict their trajectory and decay.

Most studies to date have dealt with buoyant jets rising through a quiescent ambient fluid; however, in nature most problems involve flowing ambient fluids. Relatively little experimental work has been done with buoyant jets in crossflow and, according to Hilder [Ref. 1], the trajectories of jets and the entrainment rates of ambient fluid predicted from previous work do not agree well with one another. Most mathematical models of buoyant jets in crossflow assume Gaussian profiles for velocity and temperature. Nickodem [Ref. 2] has shown through experiments that in fact, the

Gaussian profiles of velocity are altered by crossflow. This leads one to suspect that the same may be true for the temperature profiles.

The objective of this work was to develop a system to thermally map a buoyant jet in crossflow. Then, by measuring both velocity and temperature distributions, improved computations of entrainment, trajectory and heat transfer characteristics of jets can be made thereby giving rise to more accurate models.

II. BUOYANT JETS DISCHARGED TO A CROSSFLOW

A. PROPERTIES OF BUOYANT JETS

A buoyant jet is characterized by a momentum and a density differential between the jet and its surrounding ambient resulting from a variation in temperature and/or fluid concentrations. Therefore, fluid motion in the jet is governed by both inertial and buoyant forces. The non-dimensional ratio of these forces, known as the densiometric Froude number, provides an important quantitative measurement of jet characteristics and is shown below.

$$F = \frac{U_o^2}{gD(\rho_a - \rho_o)/\rho_o}$$

where U_o is the jet's discharge velocity, g is the acceleration of gravity, D is the discharge diameter of the jet, ρ_a is the density of the crossflowing ambient and ρ_o is the density of the jet fluid at its point of discharge.

The Gaussian velocity and temperature profiles assumed by most models of buoyant jets are very similar. Velocity behavior is given by:

$$U(r) = U_m \exp(-r^2/B^2)$$

where U_m is the centerline velocity, r is the independent variable and a radial distance from the centerline of the jet, and B is defined as nominal jet halfwidth. As r approaches B, velocity decays to $(1/e)U_m$ [Ref. 3]. Similarly, temperature behavior is given by:

$$T(r) = T_m \exp(-r^2/\lambda^2 B^2)$$

where T_m is the centerline temperature, r and B are defined the same as above and λ, a spreading ratio, is the inverse of the turbulent Schmidt number (S). S is defined as the ratio of the molecular momentum and mass diffusivities and is equal to ν/D_{AB} where ν is the kinematic viscosity and D_{AB} is the binary mass diffusion coefficient associated with substances A and B [Ref. 4]. Although λ varies inversely with the Froude number, the change is very slight, and in the case where substances A and B are both water, λ is slightly greater than 1. Hirst [Ref. 3] found λ to vary between 1.16 at F = 0 to 1.11 at F = infinity. The net effect then, is a more gradual temperature decay than was found with velocity.

Most buoyant jet models consider the entrainment of the ambient fluid into the jet and are based on relevant conservation equations of mass, momentum and energy. In conservation of mass, the downstream change in total mass of the jet is equated to the mass of the entrained fluid. The conservation of momentum must consider both vertical and horizontal contributions. Changes in vertical momentum are equated to the buoyant forces while changes in the horizontal momentum of the jet are equated to the horizontal momentum of the entrained fluid. The conservation of energy involves energy changes resulting from variations in the ambient temperature as caused by the jet. Hilder [Ref. 1] developed the following governing equations in non-dimensional differential form.

CONTINUITY
$$\frac{d}{ds}(u_m b^2) = 2\alpha b[|u_m - R\cos\theta| + a_3 R\sin\theta]$$

HORIZONTAL MOMENTUM
$$\frac{d}{ds}(u_m^2 b^2 \cos\theta) = 4R\alpha b[|u_m - R\cos\theta| + a_3 R\sin\theta]$$

VERTICAL MOMENTUM
$$\frac{d}{ds}(u_m^2 b^2 \sin\theta) = \left(\frac{\rho_a - \rho_m}{\rho_{ao} - \rho_{mo}}\right) \cdot \frac{2\lambda^2 b^2}{F^2}$$

ENERGY
$$\frac{d}{ds}\left(u_m T \cdot \frac{\lambda^2 b^2}{(\lambda^2 + 1)}\right) = \frac{dt_a}{ds}(u_m b^2)$$

B. FLOW REGIMES

The jet passes through several regimes as it travels from the nozzle through the ambient. The three regions most frequently referred to are shown in Figure 1. They are the zone of flow establishment, the zone of established flow and the far-field zone [Ref. 3]. In the zone of flow establishment, the velocity and turbulence profiles transform from the conditions within the nozzle to a free turbulent flow condition. It is in this region that the jet begins to mix with the ambient fluid; however, the flow is still more

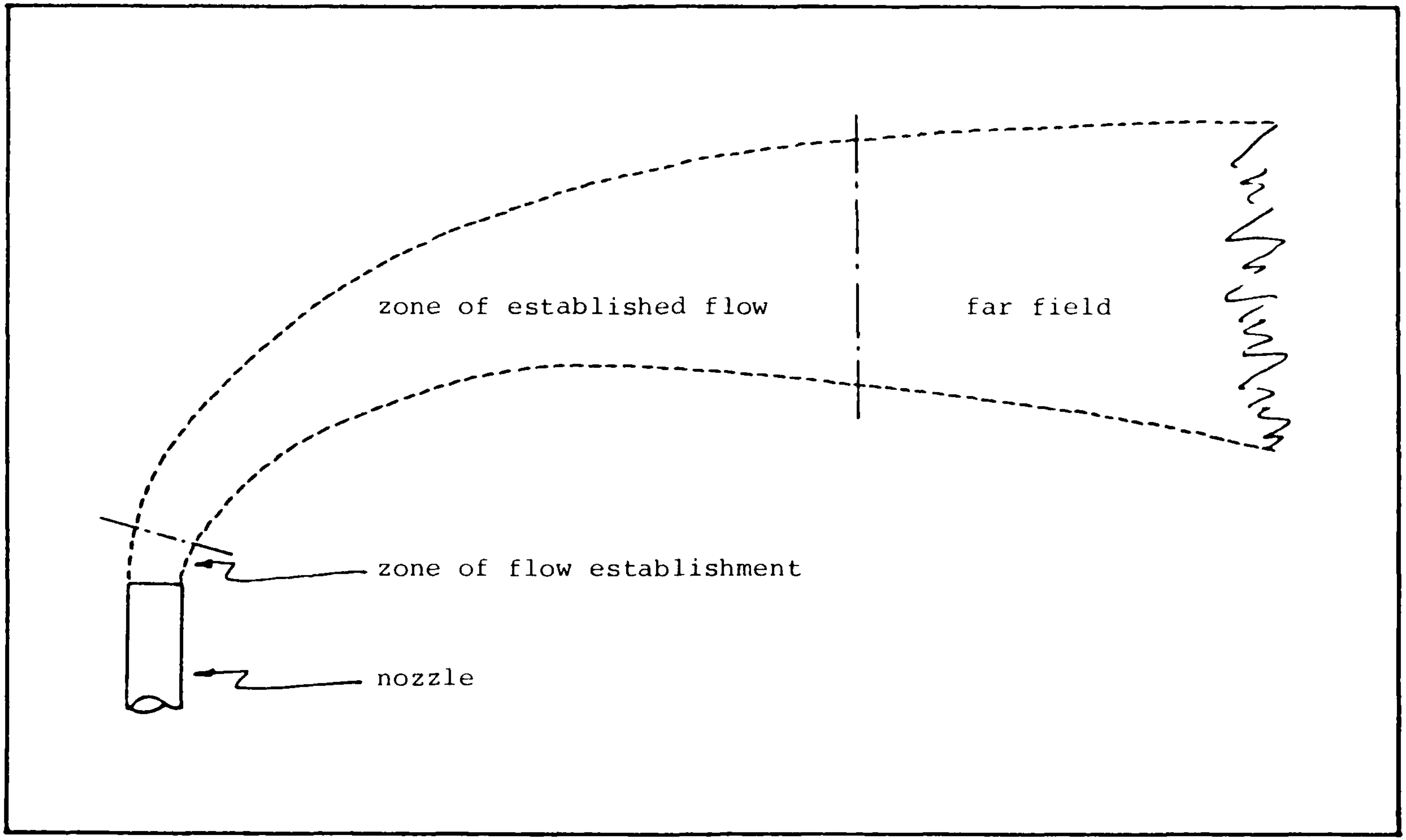


Figure 1. Typical Flow Regions in a Buoyant Jet

strongly influenced by the nozzle discharge conditions than by the ambient. When the turbulent mixing has reached the centerline of the jet, the zone of established flow is said to begin. In this region, the profiles have assumed their free turbulent shapes. Now the flow is governed by the jets' momentum and buoyancy as well as by the condition of the crossflow. The far field zone is defined as that region in which jet momentum is depleted and the jet fluid is convected and diffused by the ambient currents and turbulence.

C. EFFECTS OF CROSSFLOW

At the immediate exit of a cylindrical nozzle, a vertically discharged buoyant jet has a nearly uniform velocity distribution and has the same cross-sectional shape as the nozzle itself. The velocity gradient between the jet and the crossflowing ambient creates longitudinal shear stresses at the jet's sides, a positive pressure region immediately upstream and a negative pressure region immediately downstream of the jet. This results in the deflection of the jet's trajectory in the downstream direction (Figure 2), the creation of counterrotating vortices at the jet's outer edges and the deformation of the original circular cross-sectional shape into the form of a kidney. As the streamwise axis of the jet approaches the direction of the crossflow, these effects become progressively less pronounced.

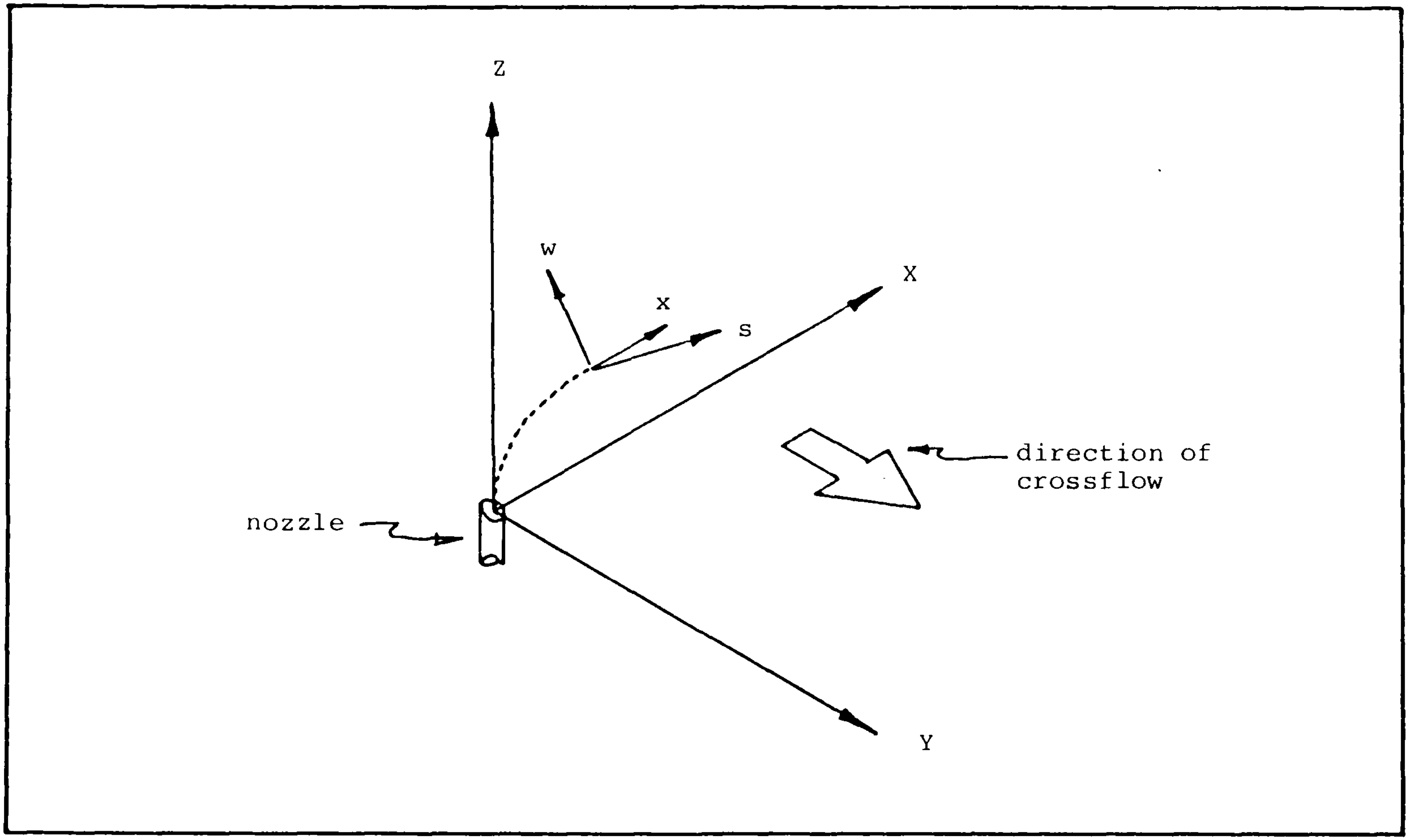


Figure 2. Coordinate System for a Buoyant Jet

III. EXPERIMENTAL APPARATUS

A. SYSTEM OVERVIEW

A surplus milling machine was configured with synchronous drive motors interfaced with a microcomputer that automatically positioned its bed. It was used as a three-dimensional positioning platform in the same manner as in the laser Doppler velocimetry work undertaken by Nickodem [Ref. 2]. The milling machine was placed adjacent to a rectangular plexiglass flume through which the crossflowing ambient fluid flowed. A vertical nozzle was installed in the base of the flume to provide the jet. A temperature probe was suspended through an opening in the top of the flume above the nozzle by an arm attached to a base mounted on the milling machine bed as shown in Figure 3. As the probe was automatically traversed through a series of preprogrammed positions across the jet, temperature data was automatically sensed and stored at high speeds by the computer.

B. CROSSFLOW SYSTEM

As illustrated in Figure 4, the crossflow circulation pump took water from the cylindrical 248.8 l (65.7 gal) reservoir shown in Figure 5 and discharged through 5.076 cm (2 in) diameter tubing into a cylindrical flow settling chamber 30.46 cm (12 in) in diameter and 60.91 cm (24 in) tall located within the 60.91 cm × 60.91 cm × 88.83 cm

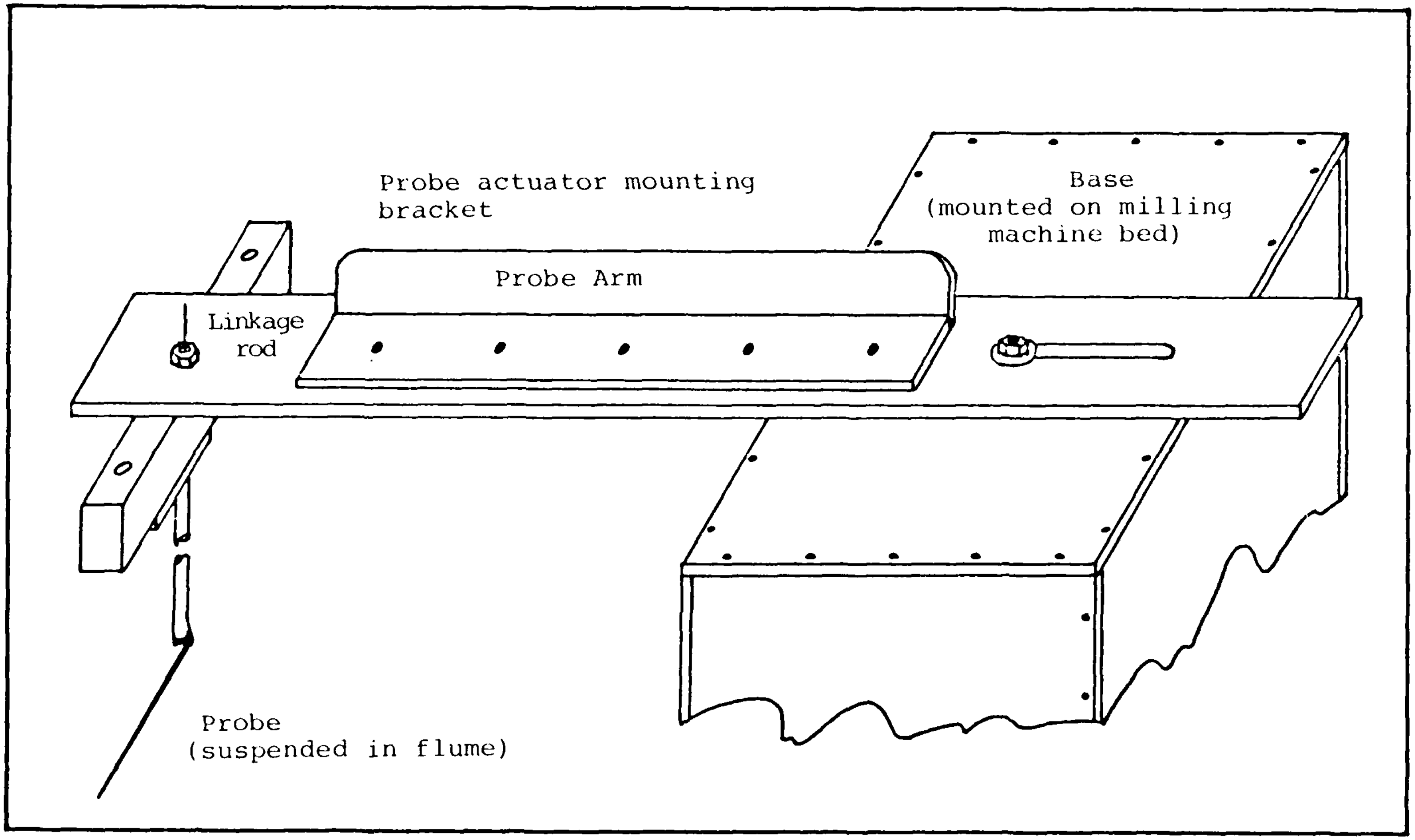


Figure 3. Probe and Traversing Mechanism

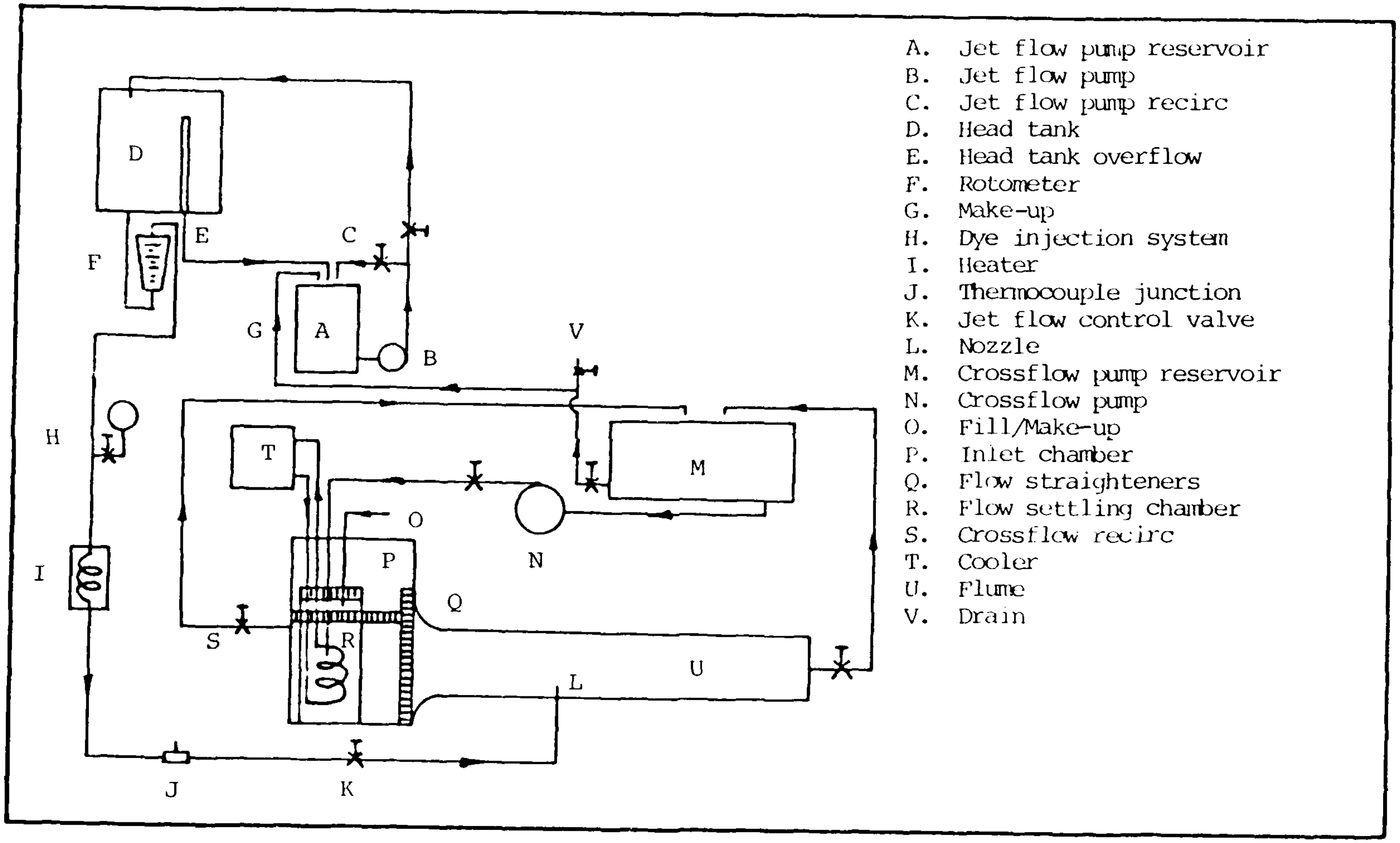


Figure 4. Crossflow and Jet Loop Piping Diagram

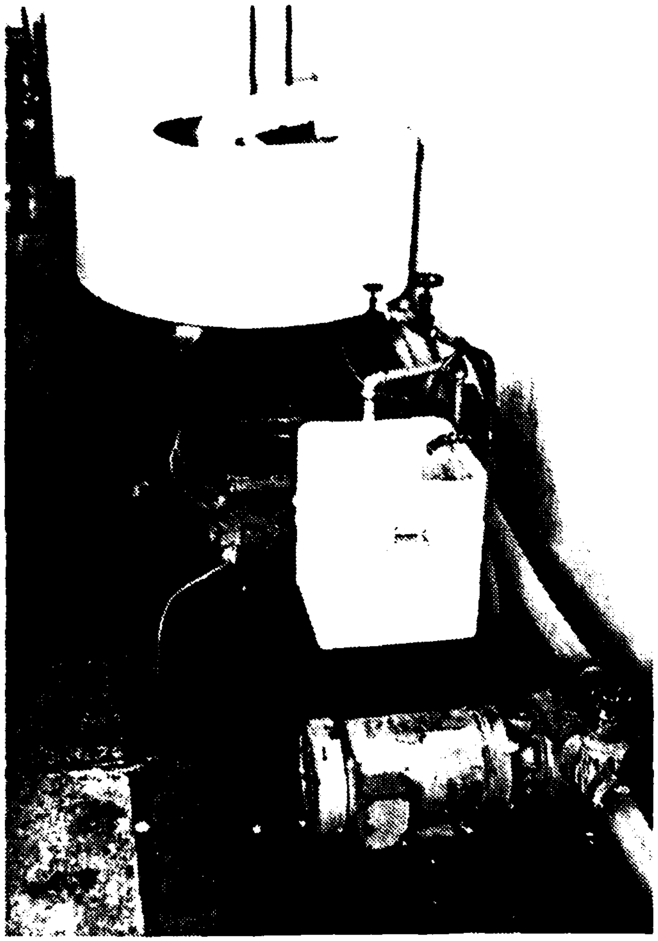


Figure 5. Flume Circulation and Jet Loop Pumps

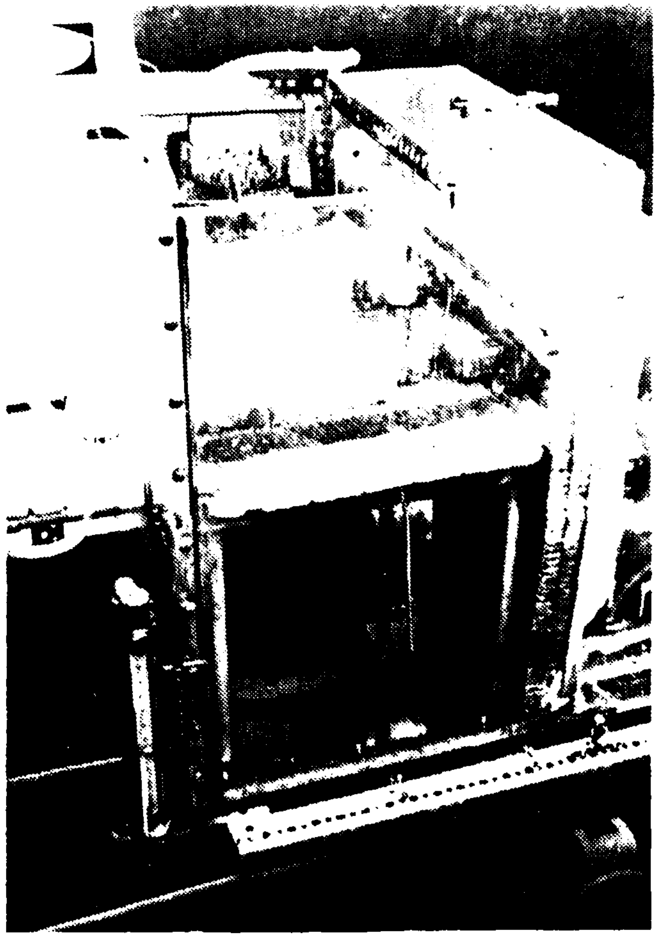


Figure 6. Inlet and Flow Settling Chambers

(24 in × 24 in × 35 in) inlet chamber shown in Figure 6. The settling chamber was sealed at its bottom so that the water spilled from its top into the inlet chamber through honeycombed flow straighteners to reduce turbulence and evenly disperse the flow. To further reduce turbulence, the flow was broken by another stack of honeycombed flow straighteners and a layer of fiberglass filter material located immediately above the normal operating water level. The flow next entered a 24.4 cm × 32.39 cm × 182.9 cm (9.625 in × 12.75 in × 72 in) flume shown in Figure 7 through a vertical section of the same honeycombed material mentioned above. To avoid inadvertent spillage over the sides of the flume during system start-up, a 5.076 cm (2 in) diameter overflow pipe was located in the inlet chamber. During normal operation, a gate valve in this piping was closed. The flow left the flume through a 7.614 cm (3 in) diameter pipe at its end and re-entered the crossflow circulation pump reservoir. A gate valve located in this piping and shown in Figure 8 was used to regulate the water level and flow velocity in the flume. The optimum adjustment of this valve was determined by trial and error to be closed two turns from its fully open position. Either a globe or ball valve would have been more appropriate for this purpose; however, neither was readily available, so the gate valve was used. The bracket shown at the base of the flume in Figure 8 maintained alignment between the flume and the milling machine. The water

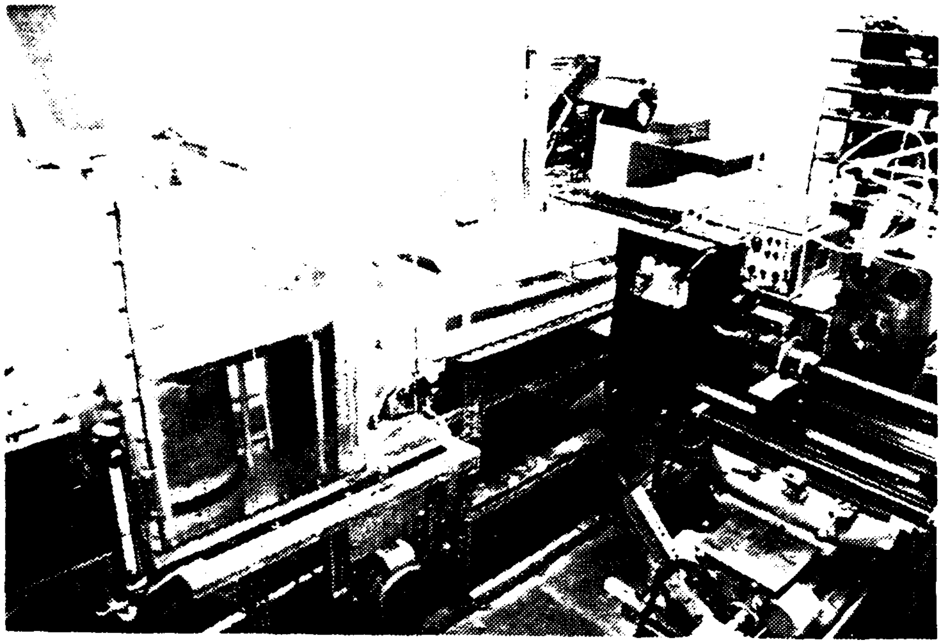


Figure 7. Flume Arrangement

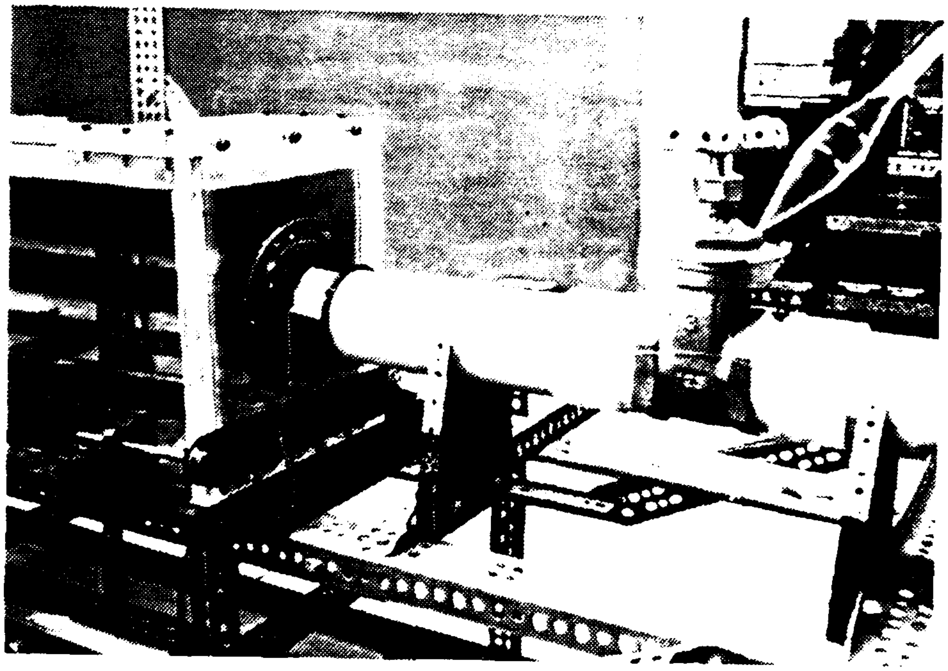


Figure 8. Flume Discharge Piping

in the flume was gradually heated by repetitive circulation through the crossflow pump and by the addition of the heated water from the jet. To maintain a constant temperature crossflow, cooling water from a refrigerated bath shown in Figure 9 was circulated through a coil of 1.269 cm (.5 in) diameter copper tubing located in the flow settling chamber. Also, fresh water was added at the flow settling chamber as an equal amount was drained from the crossflow pump reservoir through a 1.269 cm (.5 in) diameter pipe. Crossflow temperature was monitored by a Type-T thermocouple located in the inlet chamber.

C. JET SYSTEM

In reference to Figures 4 and 5, the jet flow pump circulated water from a rectangular 26.27 l (6.94 gal) reservoir and discharged through 1.26 cm (.5 in) diameter tubing to a 33.0 cm × 50.8 cm × 54.6 cm (13 in × 20 in × 21.5 in) head tank (Figure 10). The amount of flow to the head tank was regulated by a globe valve. Due to a low flow rate to the head tank, water was also recirculated back to the reservoir in order to maintain sufficient flow through the jet pump to prevent overheating it. A constant water level was maintained in the head tank by a stand pipe which allowed overflow back to the reservoir. Sufficient flow into the tank was maintained to make sure that it slightly overflowed continuously. Water drained from the bottom of the head tank through 1.26 cm (.5 in) diameter tubing and passed

Figure 9. Refrigerated Bath

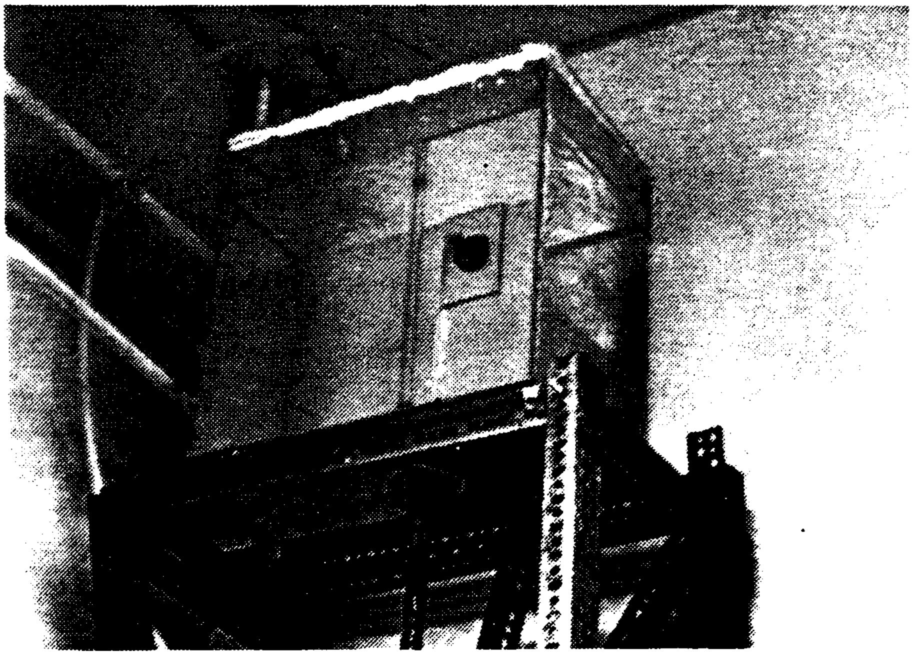


Figure 10. Head Tank

through a rotometer, a .95 cm (.375 in) tubing reducer, a dye injection system (Figure 11), a water heater (Figure 12) consisting of approximately 6.09 m (20 ft) of .95 cm (.375 in) diameter copper tubing coiled in a heated bath and finally a 7.144 cm (.28125 in) nozzle which discharged into the bottom of the flume. Flow was controlled by pinching the tubing between the heater and the nozzle with surgical clamps. Drainage from the crossflow reservoir discussed in Section III.B was used to replenish the jet reservoir. The dye injection system, used in photographing the jet, was located approximately 8.23 m (27 ft) upstream of the nozzle to minimize any disturbance to the jet that it might have caused. The majority of this distance was taken up by the heating coil mentioned above. The vertical distance between the top of the stand pipe in the head tank and the tip of the nozzle in the flume was 2.2 m (86.5 in) which equated to 21.56 KPa (3.127 psig). Jet flow temperature was monitored by a Type-T thermocouple located within the jet flow tubing approximately 1.167 m (46 in) from the nozzle.

D. TEMPERATURE PROBE

Measuring temperatures in a buoyant jet with a thermocouple is intrusive. To reduce the probability of distorting results, steps were taken to minimize the cross-sectional area of the temperature measuring device as seen by the flow of the jet. A .0254 mm (.001 in) diameter Type-E microthermocouple was selected. The suspension device for the

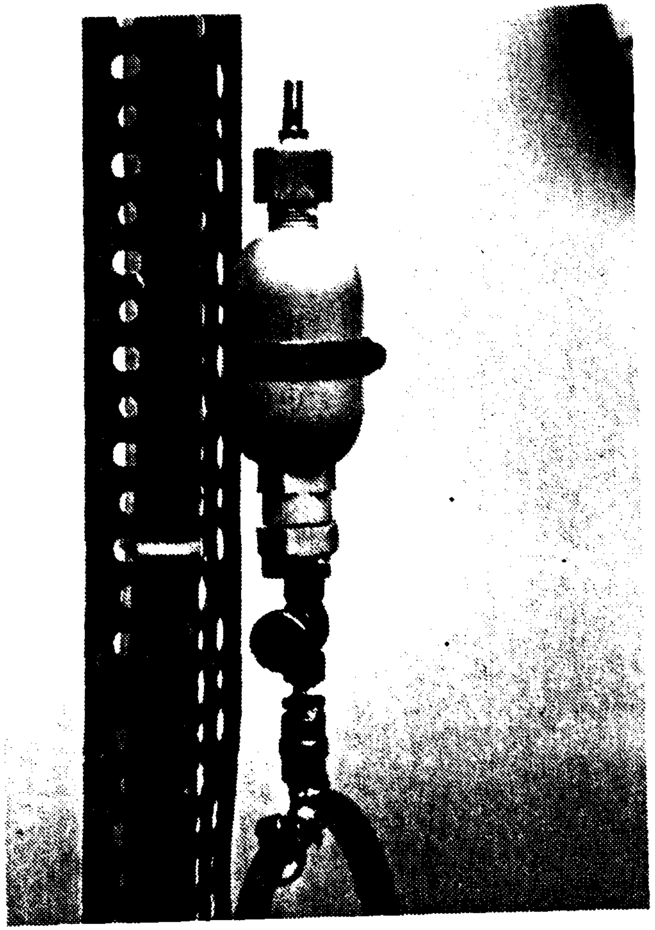


Figure 11. Dye Injection System

Figure 12. Jet Heater

microthermocouple had to be rigid and have a small cross-sectional area, for reasons discussed above, as well as be an electrical insulator to prevent interference with the thermocouple performance. A glass annulus approximately 1.45 mm (.057 in) in diameter and 11.27 cm (4.4375 in) in length was chosen. One lead of the thermocouple was threaded through the annulus and the other was glued with a fast drying modelers' glue along the outer surface, allowing the microthermocouple junction to protrude slightly from the tip of the annulus. The leads at the opposite end of the annulus were welded to .0762 mm (.003 in) diameter wire which subsequently was connected to 28 AWG extension wire to the computer. The annulus was mounted as shown in Figure 13. Henceforth, this device will be referred to as the probe.

E. PROBE ACTUATOR ASSEMBLY

The cross sectional area of the probe as seen by the jet was further reduced by orienting the probe tangentially to the trajectory of the jet as shown in Figure 14. This photograph indicated that the probe created no noticeable interference with the jet hydrodynamics. Probe orientation was accomplished by the linkage assembly shown in Figure 15. The fixed end of the probe was hinged to a streamlined tube 23.495 cm (9.25 in) long with a maximum width and depth, as seen by the jet, of 3.175 mm (.125 in) and 6.35 mm (.25 in) respectively. It was rigidly connected to the mounting

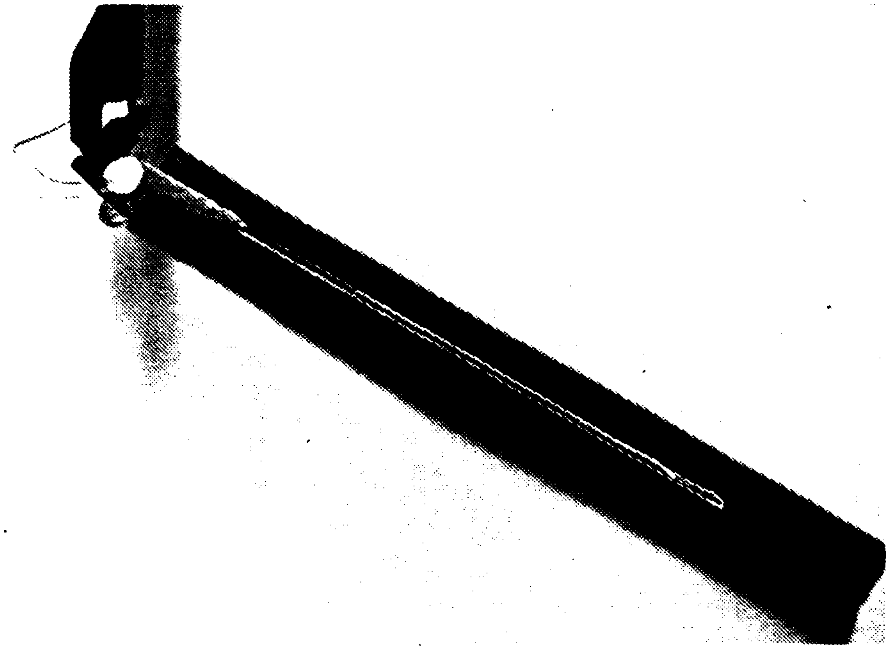


Figure 13. Probe Profile

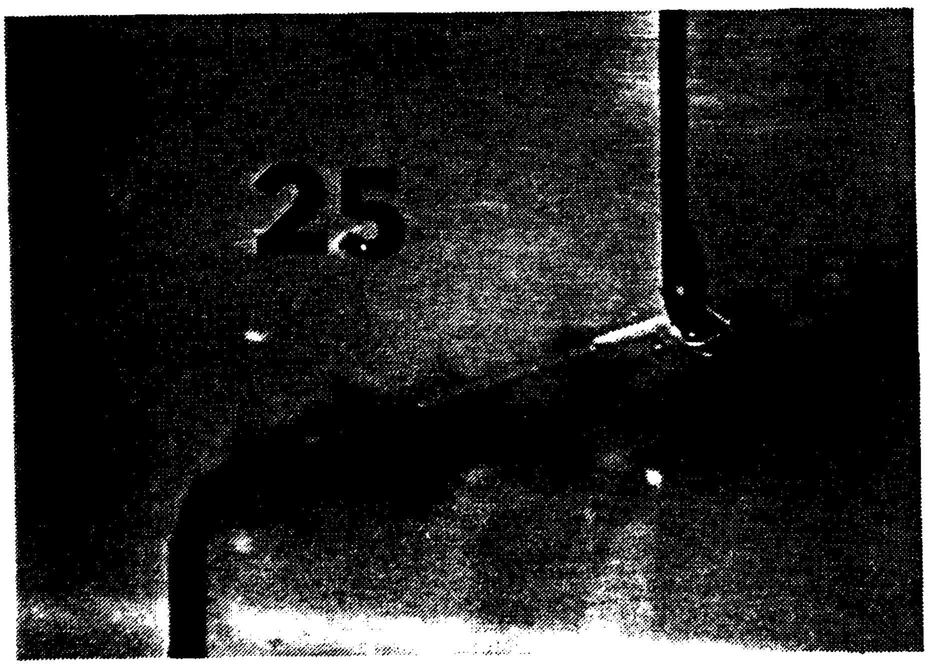


Figure 14. Probe in Jet

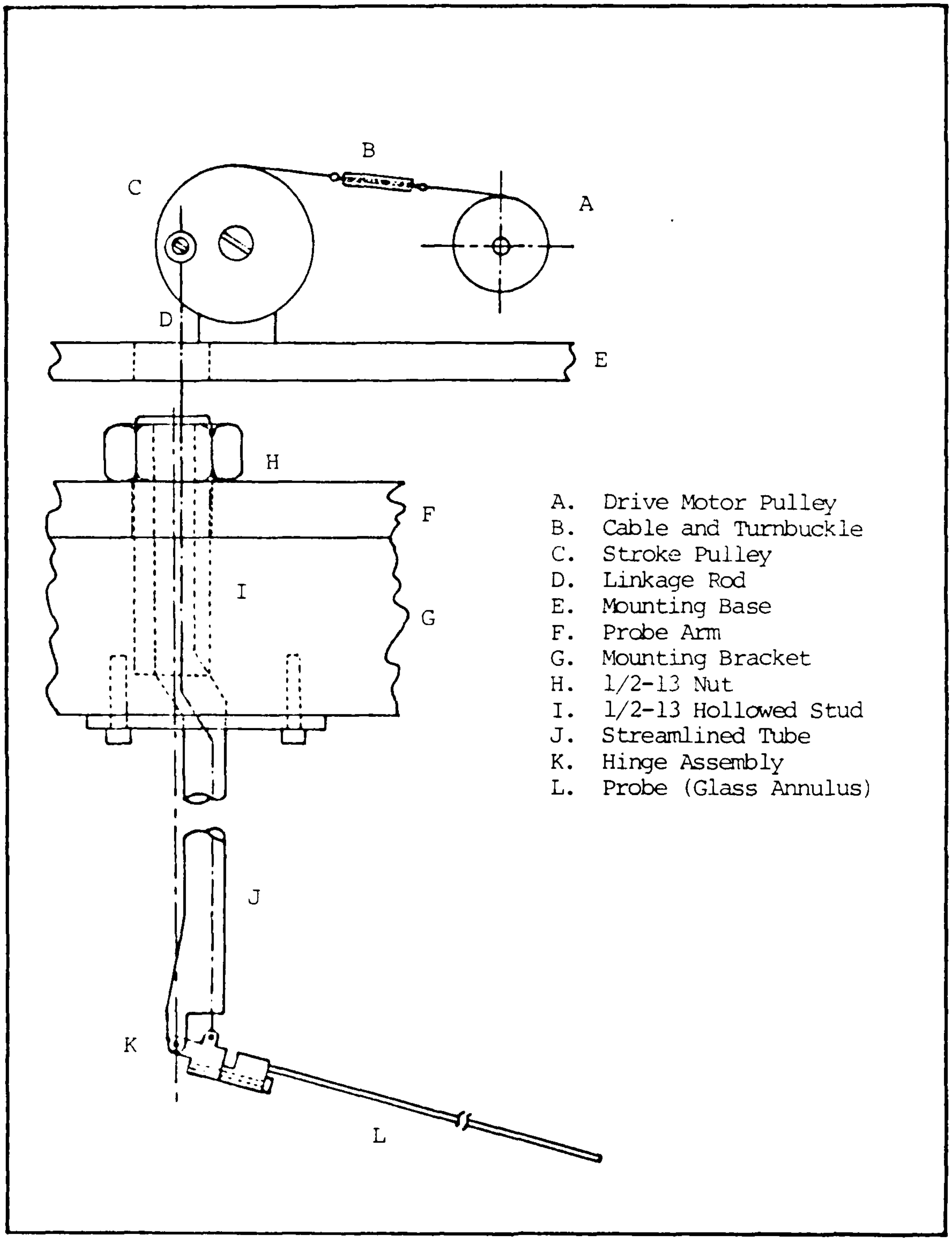


Figure 15. Probe Assembly Linkage

bracket as shown in Figure 15 which was connected to the probe arm shown in Figure 3 by a single stud which allowed pivoting of the probe from side-to-side. The stud was also hollowed so that a linkage rod could extend from the hinge assembly through the tube and stud to a stroking pulley which was rotated by a small motor. The hinge assembly and the stroking pulley were spring loaded to reduce hysteresis. As shown in Figures 16 and 17, the 1.5 VDC motor, geared to one rpm, was directly coupled to a potentiometer as well as the drive pulley. The potentiometer was configured in a voltage divider such that the amount of motor rotation, and ultimately the degree of probe deflection, was proportional to the potential difference sensed across the potentiometer. Limit switches were installed at the stroke pulley as shown in Figures 16 and 18 to prevent damage to the linkage assembly due to over-rotation.

F. MICROCOMPUTER INTERFACE

The data collection process consisted of adjusting the probe angle of deflection, traversing the three-dimensional positioning platform and measuring temperature profiles. All of the mechanisms which controlled these events were interfaced to an HP-9826 computer shown in Figure 19 through an HP-6942A multiprogrammer which performed high speed analog-to-digital conversions and ultimately provided control signals to govern relays within the system. Refer to MAIN_T in Appendix B for the microcomputer software which directed this process.

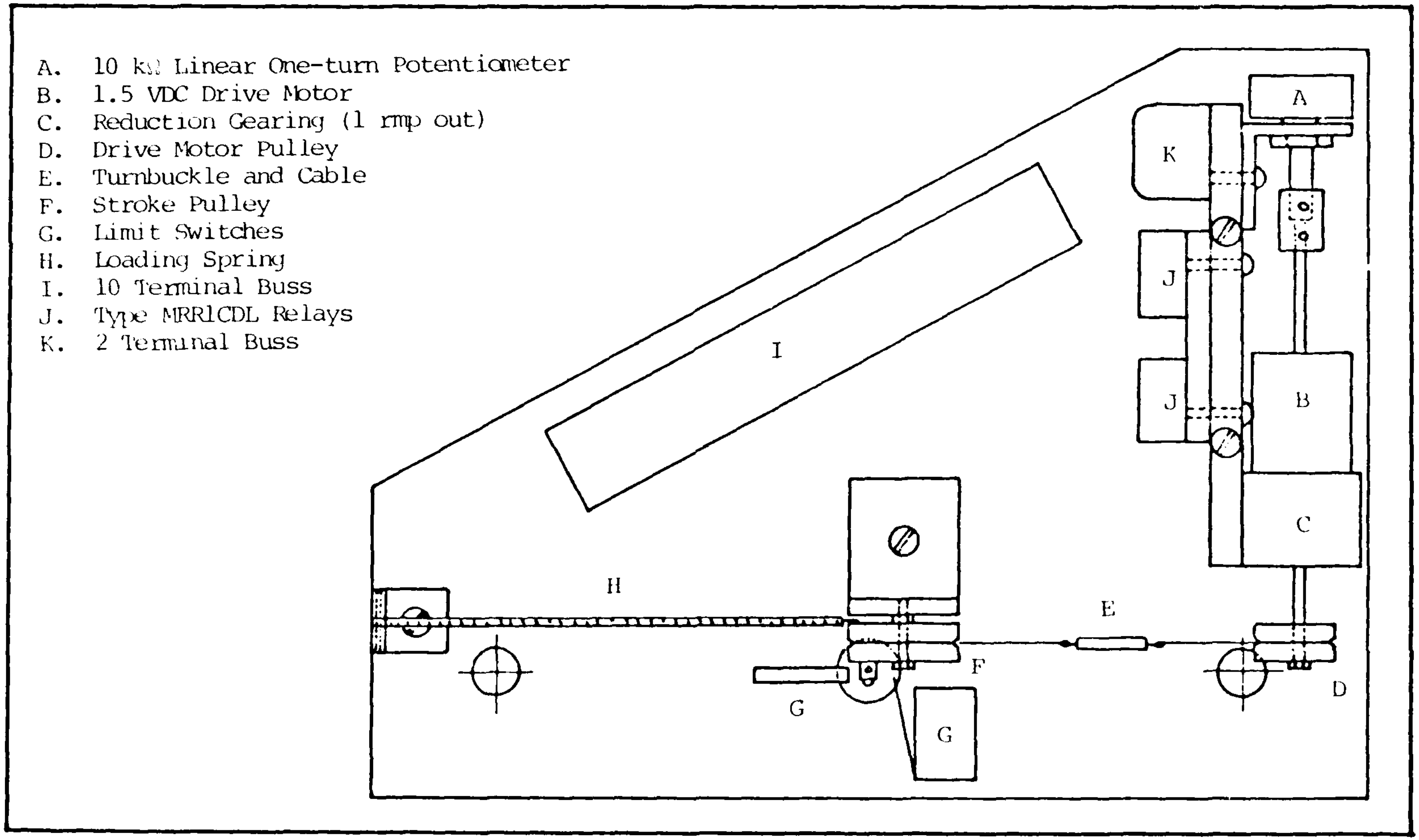


Figure 16. Probe Actuator Assembly

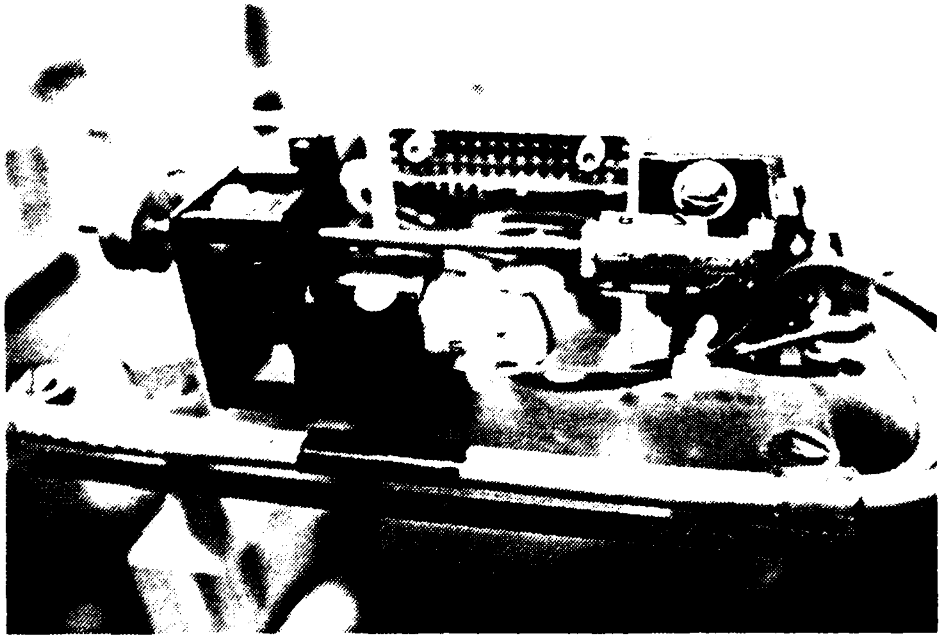


Figure 17. Probe Actuator Motor-
Potentiometer Arrangement

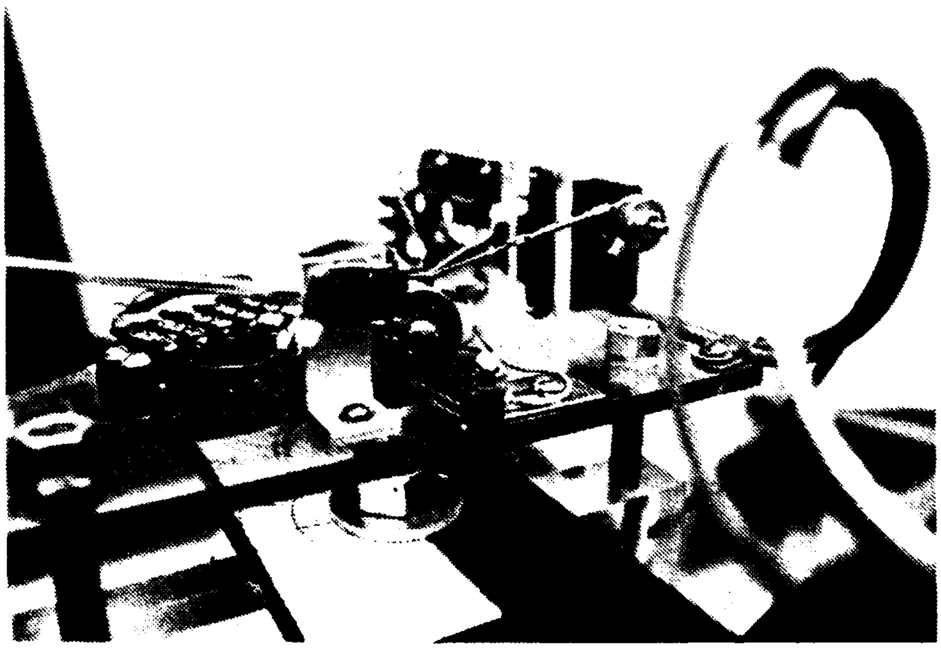


Figure 18. Probe Actuator

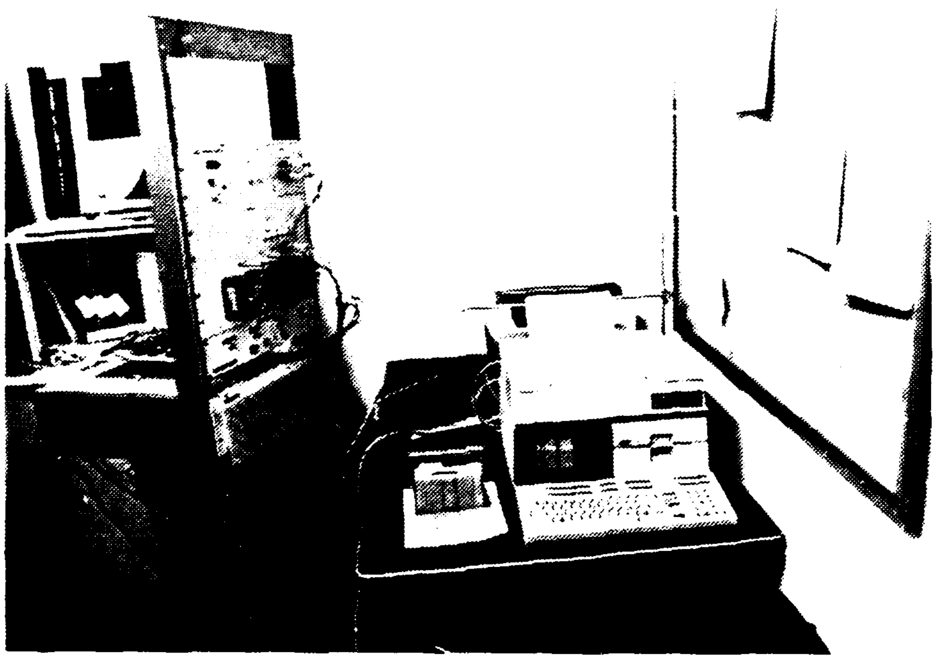


Figure 19. HP-9826 Microcomputer

1. Probe Angle Adjustment

As discussed in Section III.E, a potentiometer configured as a voltage divider provided probe angle feedback to the computer as shown in Figure 20. The direction of motor rotation was controlled by the computer through Type-MRR1CDL replays connected as shown in Figure 21. When the probe was at a desired position, a 5.0 VDC signal was applied to pins 6L and 6R which allowed both relays to assume the normally closed (NC) positions which opened the power circuit to the motor. When it was desired to rotate the motor clockwise, pin 6R was grounded which resulted in 5.0 VDC applied across the coil in the right-hand relay. This caused the relay to assume its normally open (NO) position resulting in a 1.5 VDC signal at terminal B of the motor causing it to rotate in the clockwise direction. The left-hand relay was activated in a similar manner for counterclockwise rotation. Refer to PROBE_SUBS in Appendix B for probe positioning software.

2. 3-D Positioning Platform Movement

Positioning platform movement was controlled in a manner similar to the probe and was discussed in detail by Nickodem [Ref. 2]. Refer to MTR_SUBS in Appendix B for associated HP-9826 software.

3. Temperature Data Collection

Three thermocouples were monitored in the data collection process. A Type-T thermocouple located in the inlet chamber measured the ambient fluid temperature in the flume,

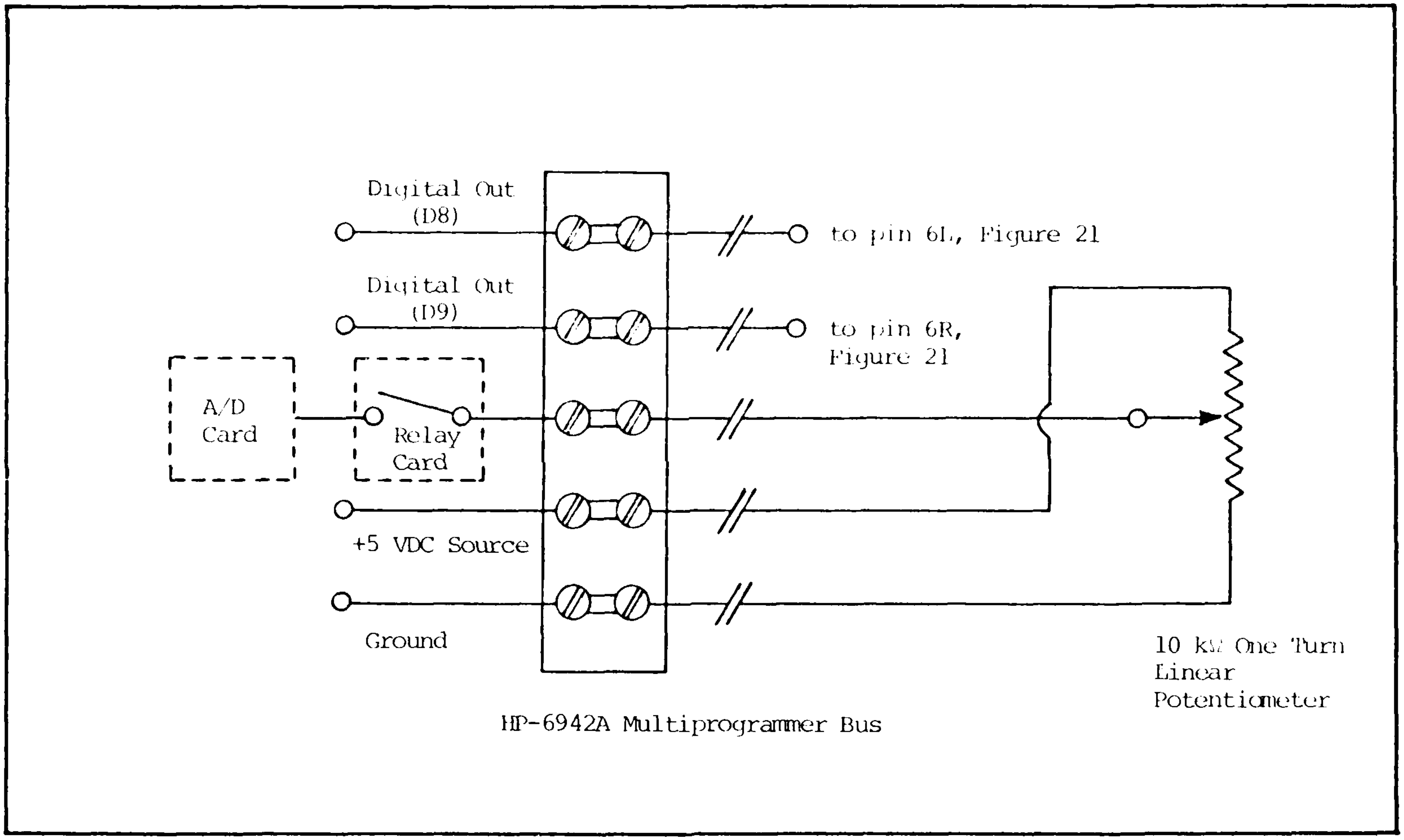


Figure 20. Computer Bus and Probe Voltage Divider Wiring Diagram

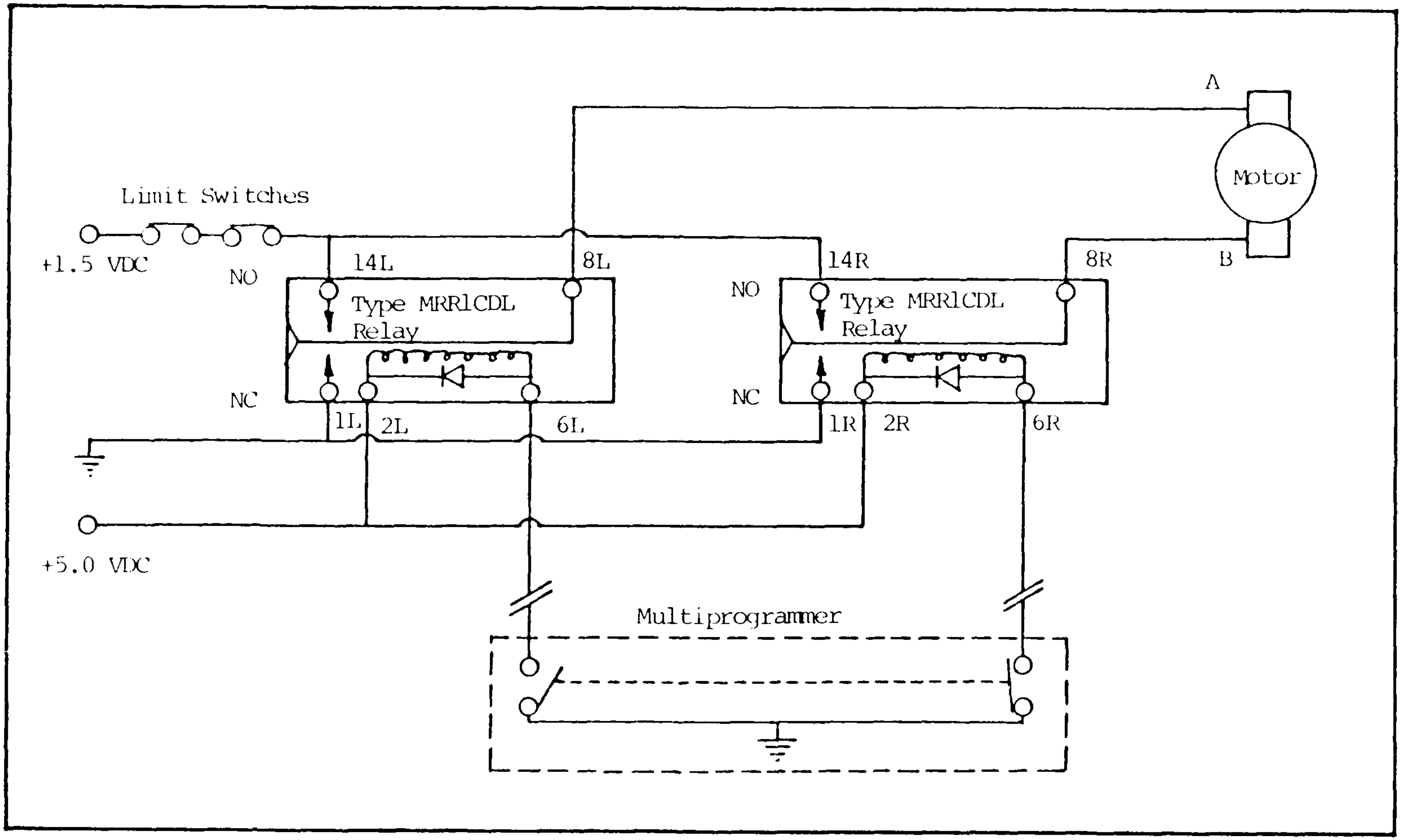


Figure 21. Probe Motor Wiring Diagram

a Type-T thermocouple located in the tubing between the heater and the nozzle measured nozzle temperature and a Type-E microthermocouple in the probe measured the temperature in the jet. The EMF's generated by these thermocouples were amplified by "Omega Omni-Amp IIB" millivolt amplifiers shown in Figure 22 prior to entering the multiprogrammer for analog-to-digital conversion and eventual transformation to temperature readings. Fourth-order least squares coefficients for this conversion were taken from Beckwith [Ref. 5]. Operation of the crossflow circulation pump created sufficient electrical interference to distort the thermocouple signals. This problem was corrected by applying a thin coating of silicon sealant to the Type-T thermocouple junctions and by connecting the crossflow circulation pump casing, the nozzle and the jet tubing in the vicinity of the nozzle thermocouple to a common ground. Because the jet tubing was plastic, it was necessary to manufacture a brass "T" connector as shown in Figure 23 which was grounded and located in close proximity to the thermocouple junction. Refer to T_SUBS in Appendix B for associated software.

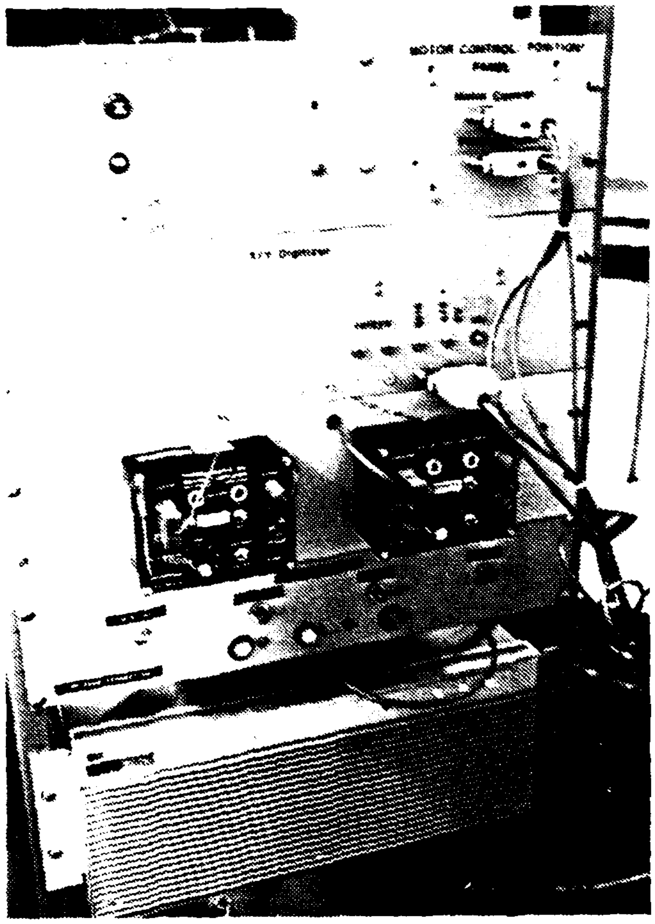


Figure 22. Thermocouple Amplifiers and HP-6942A Multiprogrammer

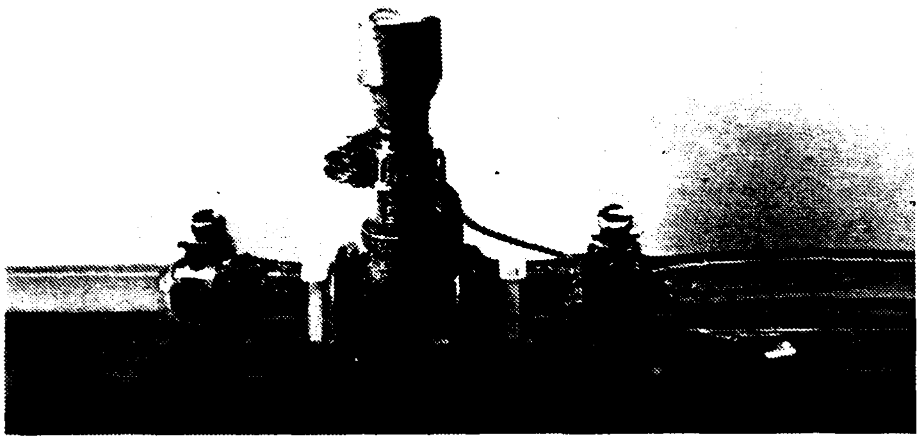


Figure 23. Nozzle Thermocouple Grounding Arrangement

IV. EXPERIMENTAL PROCEDURES

A. CALIBRATION

Before the data collection process could begin, the rotometer, the thermocouples, the probe and the positioning platform had to be calibrated. The detailed steps taken are discussed below.

1. Rotometer

With a constant level maintained in the head tank and the jet tubing disconnected from the nozzle and elevated to the same height as the top of the nozzle, five 100 ml samples were drawn through the rotometer and timed to the nearest 0.01 second at each rotometer reading from 10% to 75% in 5% increments. Flowrates and standard deviations in ml/s are shown in Table 1.

2. Thermocouples

Since nozzle and probe temperatures were to be normalized by the ambient temperature, the nozzle and probe thermocouples were calibrated relative to the ambient thermocouple by using the microcomputer program T_CAL in Appendix B. The procedure followed is outlined in the initial comments of the program. Coefficients for first order curve fits were solved by the least squares method with the mainframe programs TCAL and TFIT found in Appendix C.

3. Probe

As the probe assembly was being developed, it was convenient to test its suitability with the probe calibration panel shown in Figure 24. Resistance changes across the potentiometer were recorded for varying degrees of deflection. Analysis of this information led to improved designs from the standpoint of reduced hysteresis and repeatability. The microcomputer program PROBE_CAL in Appendix B was developed to enable calibration of the final design after it was installed in the system as shown in Figures 25 and 26. The calibration procedure is outlined in the preliminary comments of the program.

4. 3-D Positioning Platform

The positioning platform was calibrated in a manner that placed the tip of the probe at desired locations within the flume relative to the tip of the nozzle. Referring to the coordinate system illustrated in Figure 2, the center of the nozzle was defined as (0,0,0) in xyz-coordinates. The following relationships apply to the probe geometry shown in Figure 27:

$$X(\text{real}) = X_o - R_a \cos\alpha \tan(\pi/4 - \alpha/2)$$

$$Y(\text{real}) = Y_o + r_p(1 - \cos\gamma) + R_a \cos\alpha$$

$$Z(\text{real}) = Z_o - r_p \sin\gamma$$

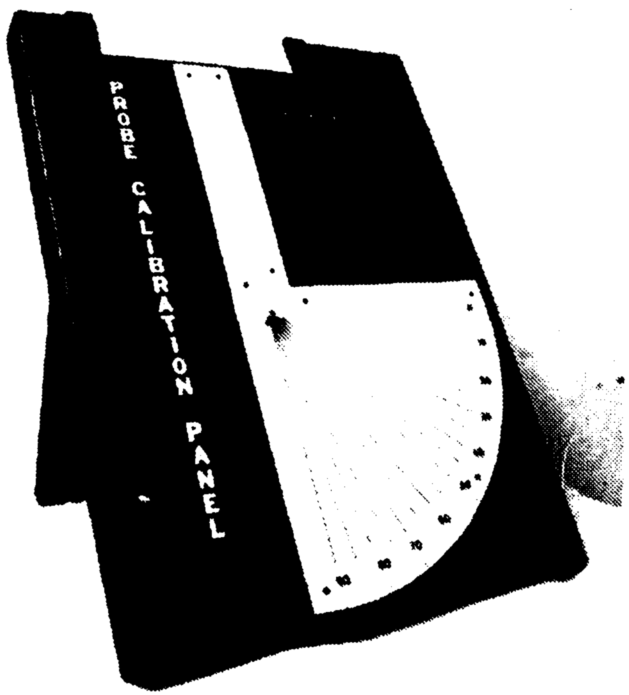


Figure 24. Probe Design Test Panel

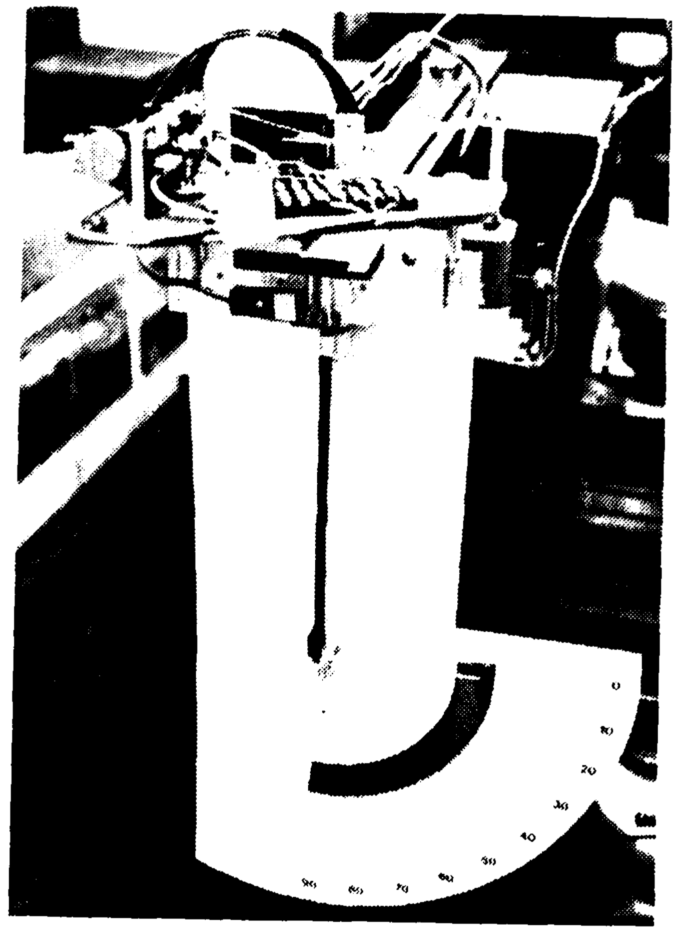


Figure 25. Probe Calibration Panel

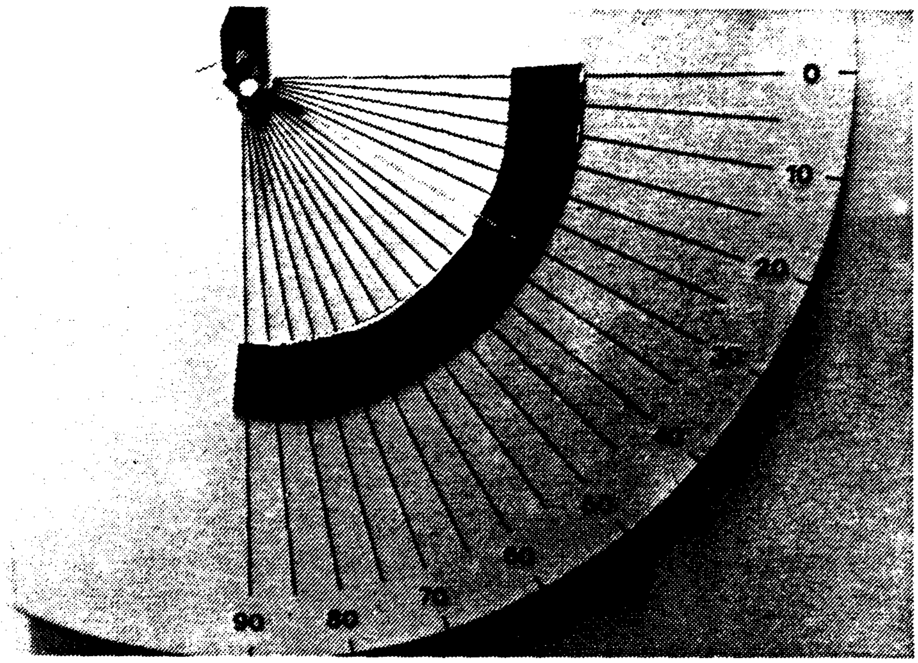


Figure 26. Probe Calibration Panel

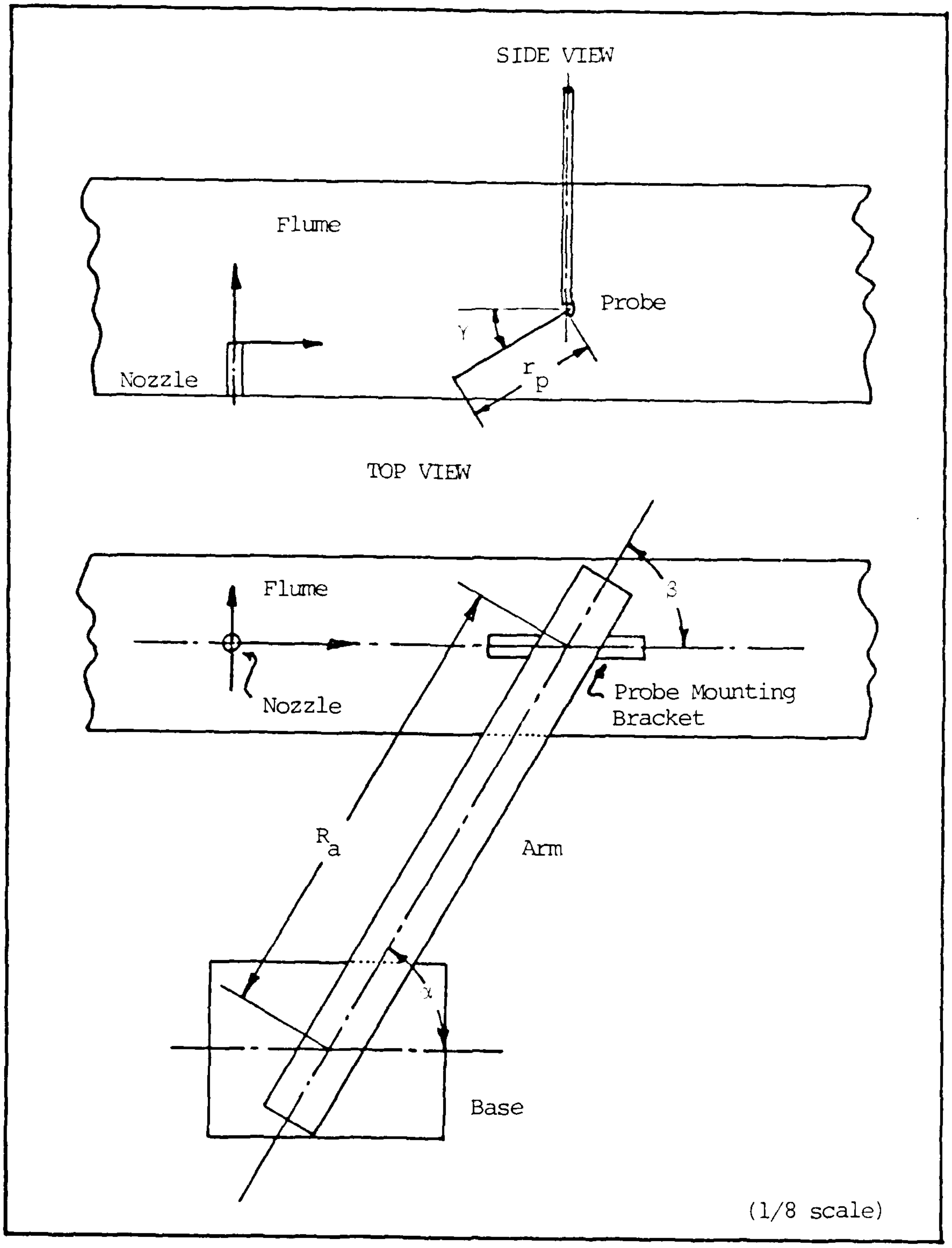


Figure 27. Probe Positioning Geometry

(X_o,Y_o,Z_o) was the position of the milling machine bed when the tip of the probe was at position (0,0,0) with γ = 0 and α = β = 90 degrees. For calibration, the probe and probe arm were configured with these settings as shown in Figure 28. With measured values of r_p and R_a entered into the microcomputer program MAIN_T, the calibration was accomplished by the program MOTOR_CAL. The step-by-step procedure followed was outlined in the subprogram "SUB Calibrate." As illustrated in Figures 3, 27 and 28, the length of R_a could be modified to compensate for adjustments of α and β to positions other than 90 degrees. Decreasing α increased the distance along the Y-axis in which the probe could be positioned. Increments of α and β were scribed on the top of the base and at the tip of the probe arm in Figure 28 to accommodate this change, if desired. The program MAIN_T queried the user for the value of α and assumed α = β. The calibration software also established position limits to prevent driving the probe into the sides of the flume.

B. PRELIMINARIES

Crossflow velocity was determined by injecting blue food coloring into the flow and timing its travel through a 1.0 m interval. The average of several trials indicated the velocity was .130 m/s (.427 ft/sec) with the flume outlet valve closed two turns from its fully open position.

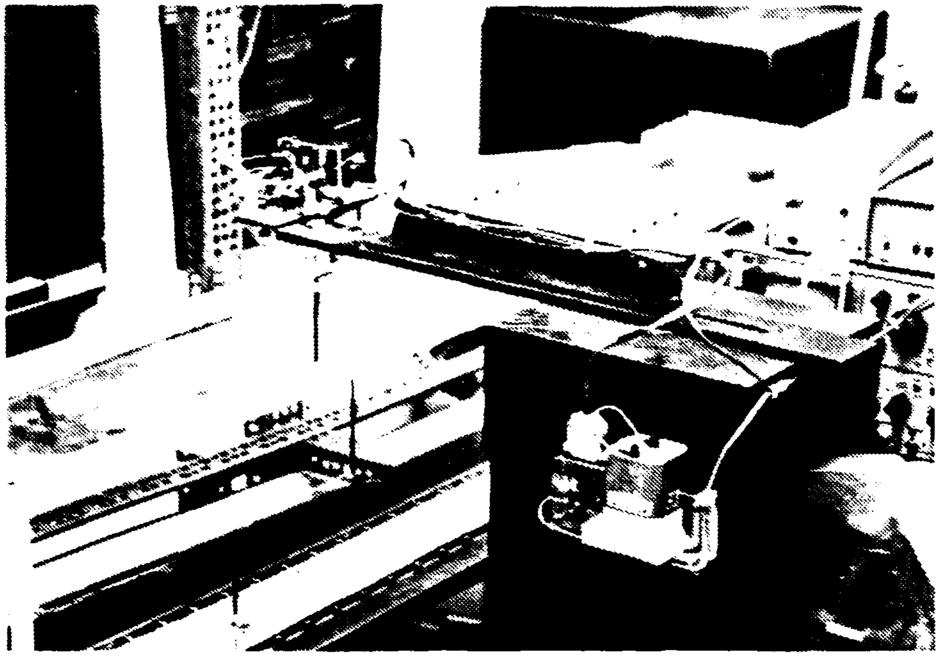


Figure 28. Probe Arm Positioning for Motor Calibration

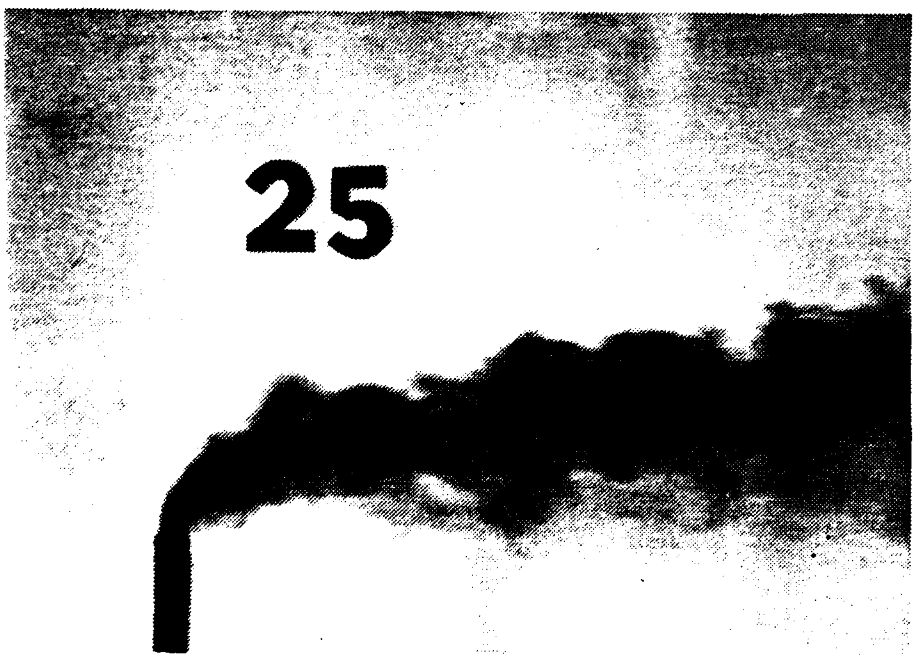


Figure 29. Typical Buoyant Jet as Obsered with Dye Injected

Photographs of jet profiles as shown in Figure 29 were taken to determine jet trajectory and halfwidths along the streamwise axis. This was done by injecting blue food coloring into the jet flow as discussed in Section III.C. The specific gravity of the food coloring was found to be considerably less than that of water. To eliminate the added buoyant effect this would have had on the jet, a small quantity of alcohol was mixed with the food coloring as suggested by Merzkirch [Ref. 6]. The amount of alcohol added was determined by trial and error. As small quantities were added and mixed, samples were gently placed on the surface of a beaker of water. Pure food coloring laid on the surface and very slowly mixed with the water. As alcohol was added, this buoyant effect grew progressively less and the mixture would settle into the water. The mixture was considered satisfactory when it no longer laid on the surface, but settled to some equilibrium position in the beaker.

Slide photographs of the jets were projected onto large sheets of 3.175 mm (.125 in) grid graph paper and digitized along approximate streamwise axes and half-width trajectories. A scaling factor was determined by equating the projected width of the nozzle to its known outer diameter of 7.9375 mm (.3125 in). The above data was fit to the Michaelis-Menter Equation [Ref. 7] shown below by the least squares method with the mainframe program JETCURV in Appendix C:

$$Z = \frac{ay}{b + y}$$

Correlation coefficients close to 1 were consistently obtained. To determine positions within the jet at which to make temperature measurements, five evenly spaced positions per jet flow rate were selected along the streamwise axis in the zone of established flow. Data planes slightly larger than the jet width were centered at these points and oriented perpendicular to the streamwise axis. One hundred data points were selected in a symmetric square matrix with points most densely populated near the center. The planes were identified alphabetically and in consecutive order from "A" to "F", where "A" represented the plane nearest the nozzle. The positions were entered into the microcomputer and stored by plane on a floppy disk by the program LOAD_XYZ in Appendix B. Accompanying each data point was a probe deflection angle used to orient the probe parallel to the path of the jet to minimize interference. This angle was determined by evaluating the first derivatives of the equations developed for the streamwise axis and the half-widths and performing an interpolation based upon the data points' position relative to the two curves.

C. DATA ACQUISITION

The flow systems were placed into operation and the ambient and nozzle temperatures were monitored with the program T_SUBS to evaluate system stability and readiness for data acquisition. The system usually took approximately two hours to come into equilibrium. This could be monitored by watching the jet nozzle temperature. When conditions were stable, data consisting of two hundred probe, ten ambient and ten nozzle temperature samples per position was collected, one plane at a time, by the program MAIN_T. The following information was stored on a floppy disk for each data point: x, y, and z coordinates; mean probe, nozzle and ambient temperatures and the standard deviation of the probe measurements. The data was transferred to the mainframe computer by using a modem, the microcomputer program SEND_DATA and the mainframe program GRAB.

D. DATA REDUCTION

The raw data was organized into a more usable format by the mainframe program TDATA which also converted the XYZ coordinates into the XSW system shown in Figure 2. The resulting data was selectively sent to the program CONTOUR4 which applied calibration coefficients to the temperature data and normalized it in the following manner:

$$T = \frac{T_p - T_a}{T_n - T_a}$$

where T_p was the jet temperature as measured by the probe, T_a was the ambient fluid temperature and T_n was the temperature of the jet within the nozzle.

Contour plots of this information, generated by the CONTOUR option of the graphics package DISSPLA [Ref. 8], are presented in Figures 30 through 35.

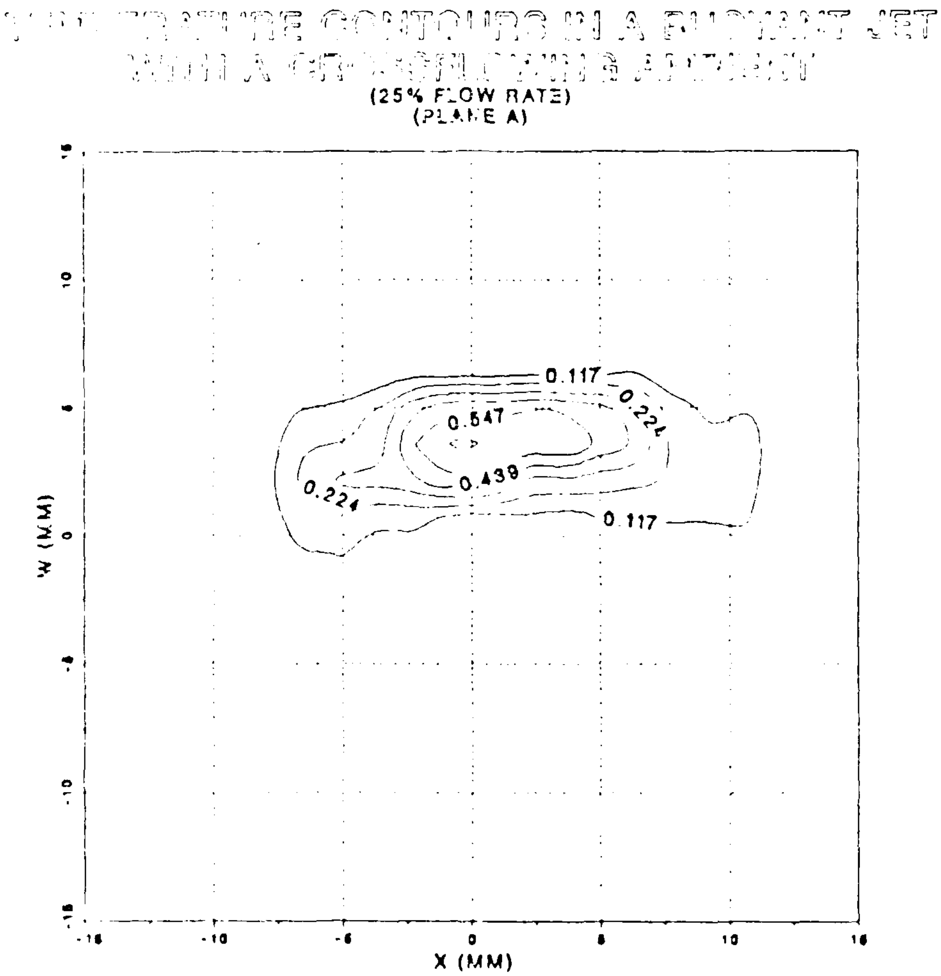


Figure 30. Plane A Temperature Contour Plot
(large scale)

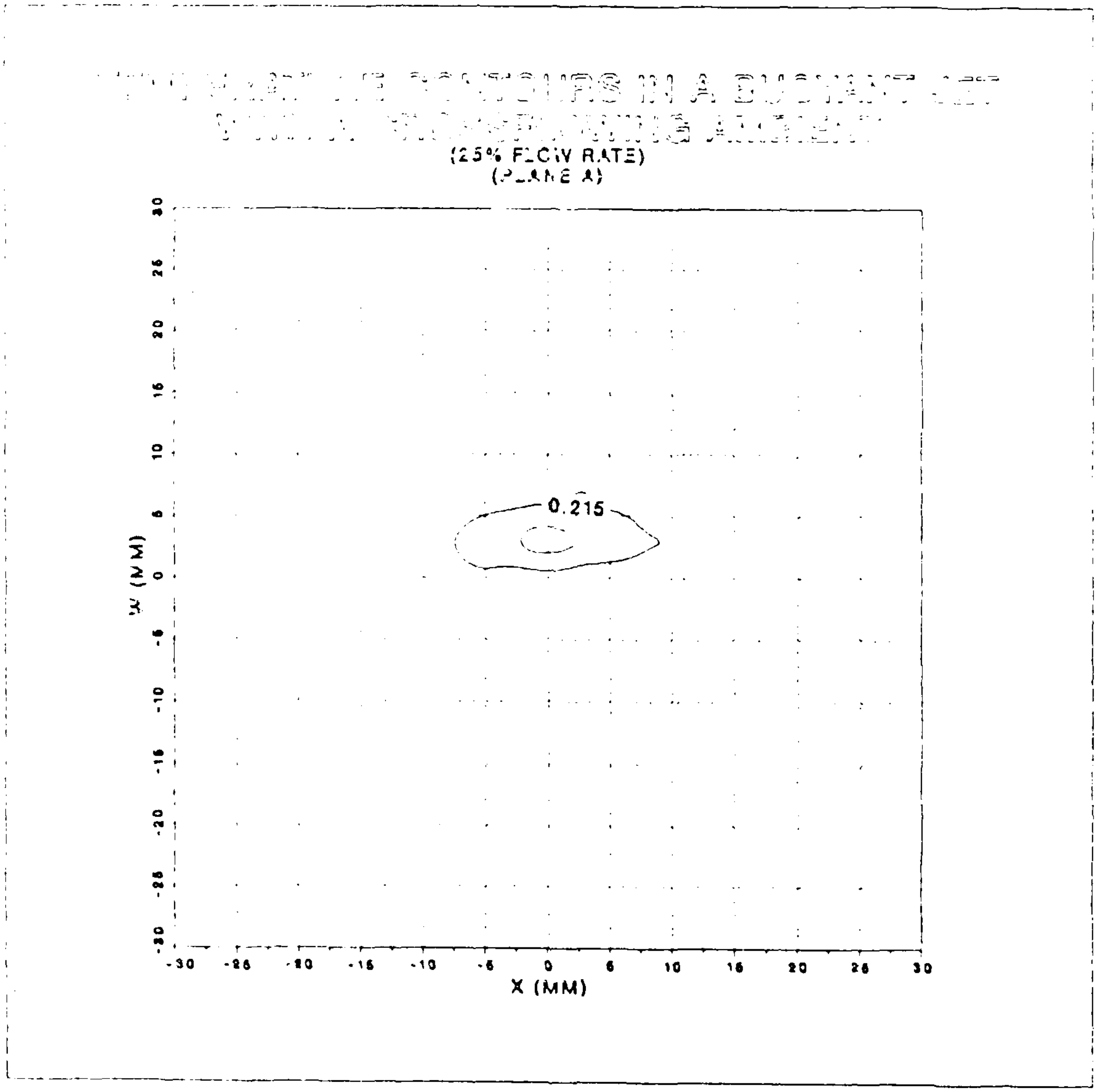


Figure 31. Plane A Temperature Contour Plot
(small scale)

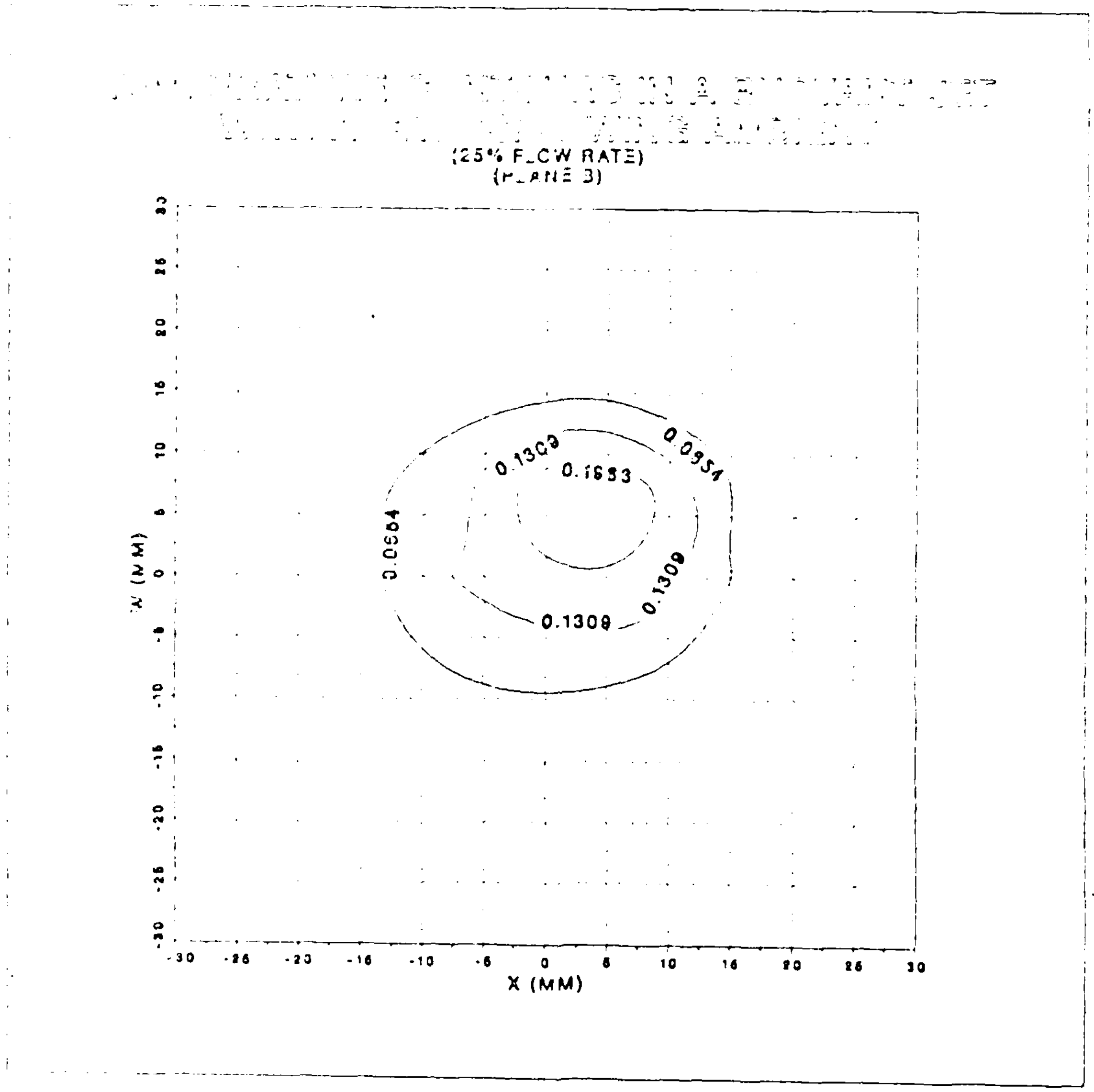


Figure 32. Plane B Temperature Contour Plot

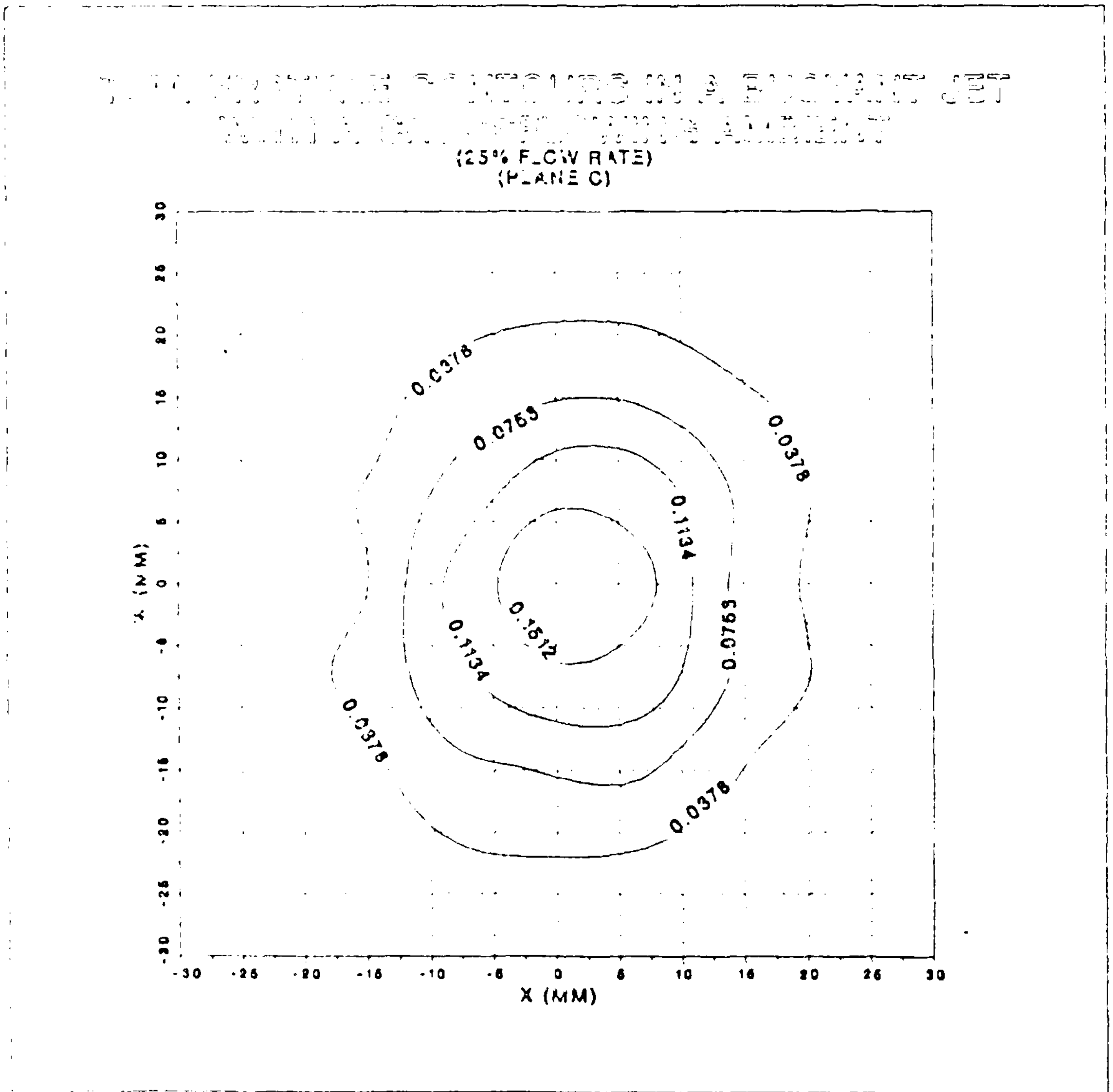


Figure 33. Plane C Temperature Contour Plot

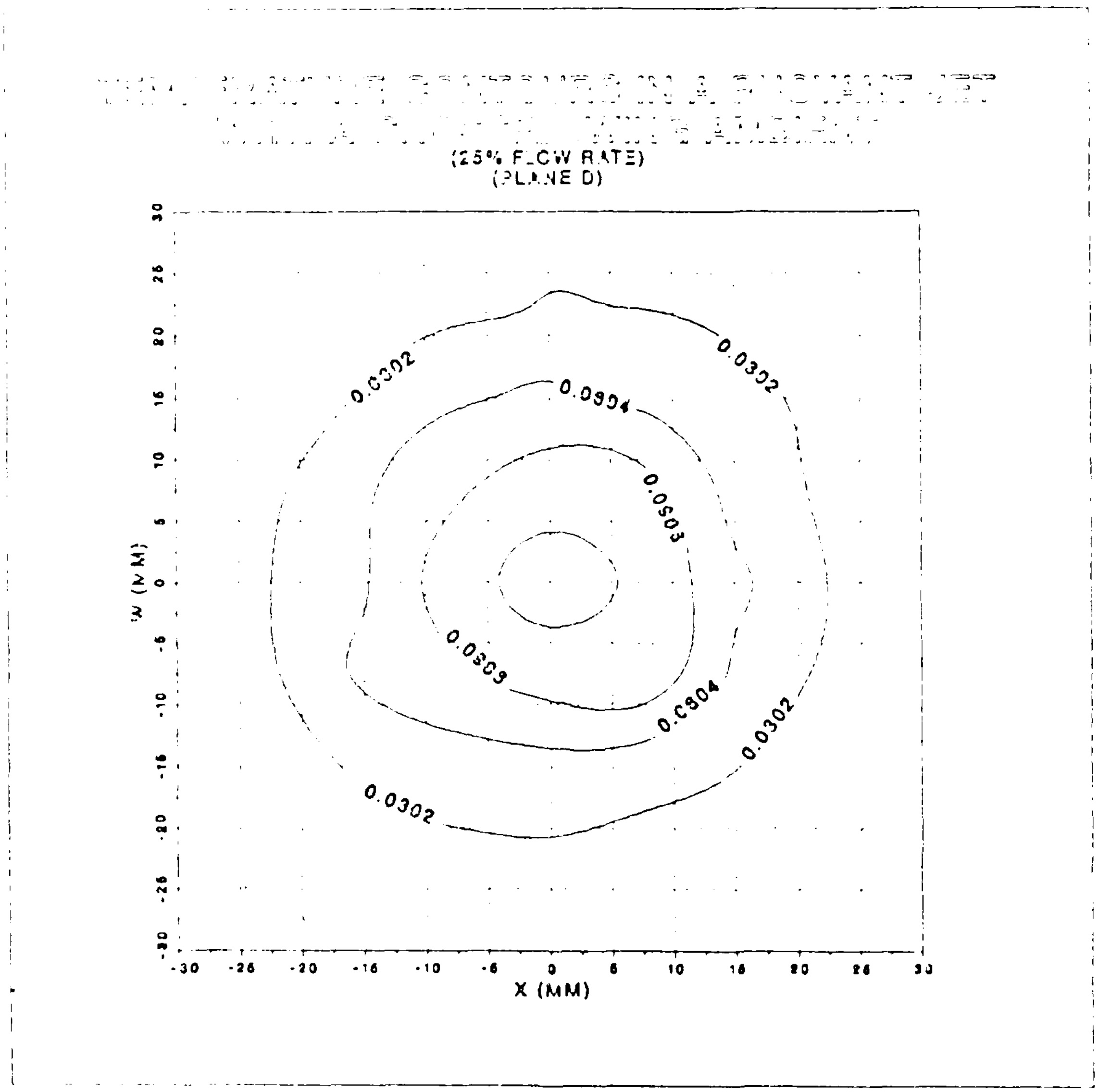


Figure 34. Plane D Temperature Contour Plot

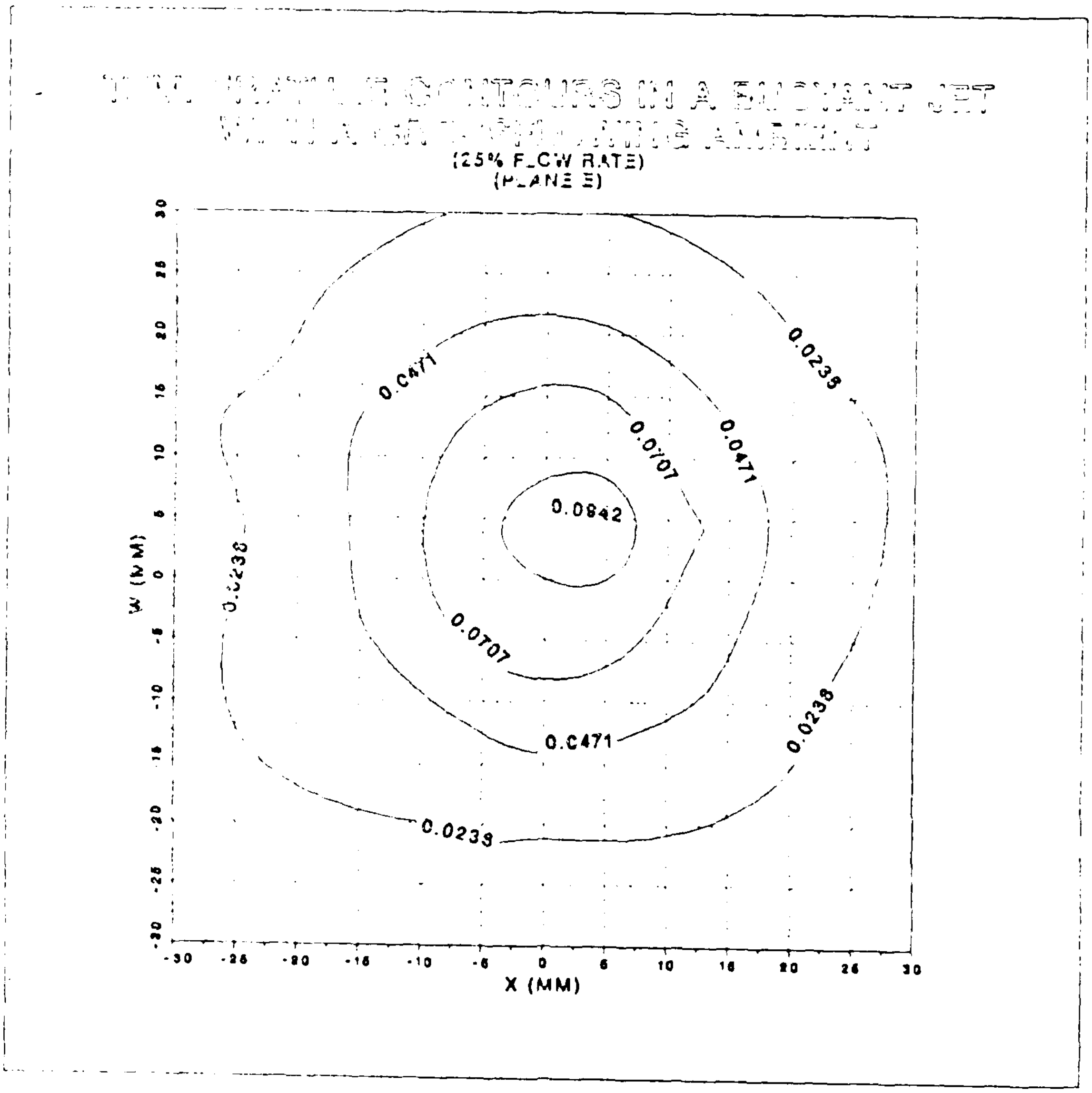


Figure 35. Plane E Temperature Contour Plot

E. RESULTS

The contour plots found in Figures 30 through 35 support the conjectures in Section II.C concerning the effects of the crossflow. Figure 30, a plot of plane A centered on the streamwise axis 46 degrees from horizontal and 7.327 mm (.288 in) downstream of the nozzle shows significant distortion. Figures 31 through 35 show planes A through E sequentially plotted on the same scale in order to observe overall jet behavior. Plane E was located 87.313 mm (3.438 in) downstream of the nozzle and 86 degrees from horizontal. It can be seen that as the jet traveled further downstream, the distorting effect grew progressively less, as expected.

The rate of heat transfer from the jet to the ambient was calculated for each plane utilizing the temperature distribution matrix generated in the program CONTOUR4 in Appendix C and the following relationship:

$$\dot{Q} = \sum_{i=1}^{m} \sum_{j=1}^{n} \rho \, c_p \, A_{ij} \, U_{ij} \, T_{ij}$$

where ρ was the relative density of the jet, A_{ij} was the area of each matrix segment, U_{ij} was the velocity in each segment, c_p was the specific heat of the jet and T_{ij} was the temperature in each segment. Velocity was measured along the streamwise axis by a laser Doppler velocimeter (LDV). It appeared to be nearly constant in the region of the jet observed. The mean and standard deviation was 44.875 mm/s and 1.8 mm/s respectively. By assuming the Gaussian profile shown in Section II.B, velocity was determined at radii corresponding to each segment in the matrix mentioned above. The rates of heat transfer are shown in Table 2.

V. CONCLUSIONS AND RECOMMENDATIONS

The objective of this thesis was to develop a computer-aided data acquisition system and construct a microthermocouple probe to be used by follow-on students to study temperature distributions in turbulent buoyant jets. Sample data was taken to verify system operability. Based on results, the system performed in a satisfactory manner and will be an invaluable tool for subsequent studies of buoyant jets in a crossflowing ambient.

Data point positions were hand calculated and loaded into the microcomputer with the program LOAD_XYZ. This was an extremely time consuming task and distracted the user from defining more than 100 data points per plane. The system can be greatly improved with the addition of microcomputer software that would automatically determine and load data point positions. Data point population could then be increased with ease which should result in smoother contour plots and more accurate heat transfer calculations.

It was necessary to continuously add fresh water to the crossflow system as an equal amount was drained from it in order to maintain the crossflowing ambient at a constant temperature. This was because the cooling coil located within the flow settling chamber was inadequate to compensate for the heat added to the system by the jet and the crossflow circulation pump by itself. Although it was possible to maintain the ambient constant within 1.4 C by this method, in the interest of conserving water, it is recommended that either a larger capacity chilled water bath or a cooling system that circulates a refrigerant rather than water be appropriated for this purpose.

APPENDIX A

UNCERTAINTY ANALYSIS

Experimental uncertainty was analyzed in accordance with the guidelines set forth by Holman [Ref. 9]. Uncertainties in the primary measurements, based upon manufacturer specifications and/or the number of significant digits which could be read, follow:

1.	Time (rotometer calibration):	.01 s
2.	Volume (rotometer calibration):	1.0 ml
3.	Rotometer reading:	1.0 percent
4.	3-D positioning platform resolution:	.1524 mm
5.	Probe deflection angle resolution:	.5 degrees
6.	Thermocouple resolution:	.0099 °C
7.	Thermocouple time constant:	.004 s
8.	Time (crossflow velocity):	.01 s
9.	Length (crossflow velocity):	.05 m

Based on the above, the uncertainty of the flow rate of the jet was estimated to be 1.0 ml/s. Maximum uncertainty in the position of the probe's tip due to the uncertainty in deflection angle was determined to be .983 mm (.0387 in). Combined with the resolution of the positioning platform, the tip of the probe was positioned with an uncertainty of 1.135 mm (.0447 in) in each plane, or with an overall uncertainty of 1.605 mm (.063 in) in three-dimensional space. Temperature was measured at the approximate rate of 100 samples per second, well within the constraints of the thermocouple time constant. Uncertainty in the temperature measurements were governed by the resolution of the analog-to-digital converter which was .0099 °C. Uncertainty in the crossflow velocity was .00625 m/s (.0205 ft/sec).

APPENDIX B

MICROCOMPUTER PROGRAMS

1. MAIN_T

```
10   !     MAIN_T
20   !
30   !       This program coordinates the entire
40   !     data-taking evolution for measuring
50   !     temperature distributions in buoyant
60   !     jets.
70   !
80   !
90   !          1. Load all subprograms
100  !          2. Input desired positions from a
110  !             disk file of the form : "RUNEX"
120  !             (value "XX" is the run number)
130  !          3. Move the 3-D positioning
140  !             platform to each position
150  !          4. Align the thermocouple probe with
160  !             the jet streamwise axis
170  !          5. Obtain 200 temperatures at each
180  !             position and compute the mean and
190  !             standard deviation (sd)
200  !          6. Write to disk:
210  !              a. T(C)
220  !              b. sd
230  !              c. x,y,z in (mm)
240  !                 nozzle centerline is (0,0,0)
250  !
260  !      Probe and arm dimensions in inches:
270  !          a. Shortest arm length  = 20.0
280  !          b. Longest arm length   = 24.0
290  !
300  Length_probe=4.4375
310  Length_arm=20.0
320  !
330  OPTION BASE 1
340  DIM Coef(12),X(500),Y(500),Z(500),Probe_angle(500)
350  LOADSUB ALL FROM "T_SUBS"
360  LOADSUB ALL FROM "MTR_SUBS"
370  LOADSUB ALL FROM "PROBE_SUBS"
380  CALL Retrieve_coef(Coef(*),"motor_coef")
390  !
400  !     2. Input desired positions from disk
410  !
420      I=1
430      INPUT "Angle of arm relative to +Y-axis?",Angle_arm
440      INPUT "Filename for positioning data?",Filename$
450      ASSIGN @File4 TO Filename$
460 Go_on:          !
470                ENTER @File4;X(I),Y(I),Z(I)
480                !
490                IF X(I)<>-100 THEN
500                  IF X(I)=-999 THEN
510                     Probe_angle(I)=Y(I)
520                     P_angle=Y(I)
530                  ELSE
540                     Probe_angle(I)=P_angle
550                     I=I+1
560                  END IF
570                  !
580                  GOTO Go_on
590                END IF
600                !
```

```
610                 !
620                  GOTO Go_on
630               END IF
640               !
650               ASSIGN @File4 TO *
660               Nitems=I-1
670               !
680        !3. Begin loop to take data at each point
690        !
700        INPUT "NAME OF FILE WHERE DATA IS TO BE STORED?",Filename1$
710        Records=(Nitems*8*7/256)+2
720        CREATE BDAT Filename1$,Records
730        ASSIGN @File1 TO Filename1$
740        !
750        !
760        !
770        FOR I_position=1 TO Nitems
780            !
790            !a. Move milling machine and move the
800            !   thermocouple probe.
810            !
820            I=I_position
830         CALL Move_ldv_to(X(I),Y(I),Z(I),Lenght_arm,Length_probe,Angle_arm,Prob
e_angle(I),Coef(*))
840            !
850            !
860            !b. Obtain mean temperature and sd
870            !
880            CALL T_couple(T,"PROBE","C",200,St_dev)
890            !
900            !c. Measure ambient and jet temps
910            !
920            CALL T_couple(T_ambient,"AMBIENT","C",10,Sd)
930            CALL T_couple(T_nozzle,"NOZZLE","C",10,Sd)
940            !
950            !d. Write all information to disk
960            !
970            OUTPUT @File1;X(I),Y(I),Z(I)
980            OUTPUT @File1;T,T_ambient,T_nozzle
990            OUTPUT @File1;St_dev
1000       NEXT I_position
1010       !
1020           !4. Close files
1030           !
1040           OUTPUT @File1;-100
1050           ASSIGN @File1 TO *
1060           !
1070           !5. Terminate program
1080           !
1090           PRINT "All done!"
1100           BEEP
1110           BEEP
1120 END
```

2. PROBE_SUBS

```
10    !  PROBE_SUBS
20    !
30    !   This program moves the temperature probe
40    ! to desired angles of deflection.
50    !
60    !   NOTE:  Calibration coefficients are
70    !          entered in SUB Read_angle,
80    !          beginning at line 290.
90    !
100   CALL Read_angle(Angle)
110      PRINT "The probe is presently at":Angle;" degrees from horizontal."
120   INPUT "What is your desired angle for the probe?",Desired_angle
130   CALL Probe_move(Desired_angle)
140   GOTO 120
150   END
160   !
170   !
180   !
190   SUB Read_angle(Actual_angle)
200       !This program reads the present angle
210       !of the probe and returns it
220       ! 0 degrees = horizontal
230       !90 degrees = vertical, downward
240       !
250       !A0, A1 and A2 = coefficients for a
260       !second order curve fit of mV vs angle
270       !of deflection data.
280       !
290       !Coefficients for 8 June data follow:
300       !
310       A0=-23.8657824056449
320       A1=3.09995506283164E-2
330       A2=-1.22922305456149E-6
340       !                    ! FORMAT A/D CARD
350       OUTPUT 723:"CC,1T" !Clear Relay Card
360       OUTPUT 723:"CC,3T" !Clear A/D Card
370       OUTPUT 723:"CC,7T" !Clear Digital Card
380       OUTPUT 723:"SF,3,3,3,1.25,12T"
390       OUTPUT 723:"OB,1,10,1T" ! CLOSE RELAY
400       OUTPUT 723:"IP,3T"     ! START A/D
410       ENTER 72301;V
420       Actual_angle=A0+A1*V+A2*V*V
430    SUBEND
440    !
450    !
460    !
470    SUB Probe_move(Desired_angle)
480         !This subprogram moves the probe to the
490         !desired angle
500         !
510         !  0. Check to see if the angle is in
520         !     an acceptable range.
530         !
540            OUTPUT 723:"OP,1,0T" !Clear Relays
550            CALL Clear_screen
560            !
570            IF Desired_angle>90 THEN
580               BEEP 3400,[illegible]
590               BEEP [illegible]0,1
```

```
600             BEEP 3400,.1
610             PRINT "Desired angle exceeds 30 degrees!!!"
620             SUBEXIT
630          END IF
640          !
650          IF Desired_angle<0 THEN
660             BEEP 3400,.1
670             BEEP 3800,.1
680             BEEP 3600,.1
690             PRINT "Desired angle is negative!!!"
700             SUBEXIT
710          END IF
720        !
730        !  1.a. Clear the digital output card.
740        !    b. Format the A/D card.
750        !    c. Close the relay that corrects
760        !       the probe potentiometer to the
770        !       A/D converter.
780        !
790          OUTPUT 723;"OP,7,0T"
800          OUTPUT 723;"SF,3,3,3,1.25,3T"
810          OUTPUT 723;"CC,1T"
820          OUTPUT 723;"OB,1,10,1T"
830        !
840        !  2. Define the acceptable tolerance
850        !     in the angle (degrees).
860        !
870          Tolerance=.5
880        !
890        !  3. Control loop.
900        !
910 Repeat:  CALL Read_angle(Actual_angle)
920          PRINT "ANGLE =";
930          PRINT USING "DDD.DD";Actual_angle
940            BEEP Actual_angle*100,.05
950          Angle_error=(Desired_angle-Actual_angle)
960          !
970          IF ABS(Angle_error)>Tolerance THEN
980             !
990             IF Angle_error>=0 THEN
1000               Direction$="Down"
1010            ELSE
1020               Direction$="Up"
1030            END IF
1040            !
1050            CALL Motor_go(Direction$)
1060            !
1070            GOTO Repeat
1080         END IF
1090 OUTPUT 723;"OP,7,0T"
1100 OUTPUT 723;"CC,1T"  !Clear relay card.
1110 SUBEND
1120 !
1130 !
1140 !
1150 SUB Motor_go(Direction$)
1160      IF Direction$="Up" THEN
1170         Lbit=8
1180      END IF
1190      !
```

```
1200     IF Direction$="Down" THEN
1210        Lbit=7
1220     END IF
1230     !
1240     OUTPUT 723;"OP,7";2^Lbit;"T"
1250 SUBEND
1260 !
1270 !
1280 !
1290 SUB Clear_screen
1300 ! Clear the CRT.
1310 !
1320    OUTPUT 2 USING "#,B";255,75
1330    GCLEAR
1340 SUBEND
```

3. MTR_SUBS

```
10    !    MTR_SUBS
20    !
30    !  The following series of subroutines
40    ! are utilized to calibrate the positioning
50    ! platform and ultimately to move the probe
60    ! tip to desired positions within the jet.
80    !
90    SUB Draw_flume
100   !
110   ! Draw the Buoyant Jet Flume on the CRT.
120   !
130   GCLEAR
140   GRAPHICS ON
150   WINDOW 0,48,0,38
160   LINE TYPE 1
170   MOVE 6,6
180   ! Draw the top view
190   IDRAW 36,0
200   IDRAW 0,10
210   IDRAW -36,0
220   IDRAW 0,-10
230   !
240   IMOVE 6,0
250   IDRAW 0,10
260   IMOVE 24,0
270   IDRAW 0,-10
280     ! label "Top".
290     MOVE 0,10
300     CSIZE 5
310     LABEL "Top"
320   ! Draw the side view.
330   MOVE 6,22
340   IDRAW 36,0
350   IDRAW 0,13
360   IDRAW -36,0
370   IDRAW 0,-13
380   IMOVE 6,0
390   IDRAW 0,13
400   IMOVE 24,0
410   IDRAW 0,-13
420     ! Label "Side"
430     CSIZE 5
440     MOVE 0,29
450     LABEL "Side"
460   !
470   ! Label the picture.
480   !
490     MOVE 11,35
500     CSIZE 7
510     LABEL "BUOYANT JET FLUME"
520   !
530   ! Put on the nozzle.
540   !
550   MOVE 14,22
560   IDRAW 0,2
570   IDRAW .25,0
580   IDRAW 0,-2
590   !
600   IMOVE 0,-2
610   CSIZE 3
```

```
620   LABEL "nozzle"
630   !
640   MOVE 14,11
650   CSIZE 3
660   LABEL "o"
670   !
680   ! Indicate the direction of flow.
690   !
700   MOVE 8,11
710   IDRAW 2,0
720   IDRAW -.5,-.5
730   IDRAW .5,.5
740   IDRAW -.5,.5
750   !
760   MOVE 8,27
770   IDRAW 2,0
780   IDRAW -.5,-.5
790   IDRAW .5,.5
800   IDRAW -.5,.5
810   !
820   SUBEND
830   !
840   !
850   !
860   !
870   !
880   !
890   SUB Calibrate(Filename$)
900   !
910     OPTION BASE 1
920     DIM Coef(12)
930   !
940   !    Calibrate the positioners on the
950   !    milling machine movement.
960   !    Onto disk, write out the calibration
970   !    coefficients and the hard boundaries
980   !    that must be observed!
990   !    This file will be called "Motor_coef".
1000  !
1010  ! A.   Position the probe volume at the
1020  !      wall of the tip of the nozzle. This
1030  !      position is (0,0,0).  All readings
1040  !      will be in inches.  Read all three
1050  !      potentiometers.  Ask the user for the
1060  !      nozzle outer diameter and compute the
1070  !      zero position.  Ask the user for the
1080  !      milling machine readings.
1090  !
1100  ! B.   Next, move the bed to some new posi-
1110  !      tion using the override switches.
1120  !      Take readings from the pots and ask
1130  !      for the milling machine readings.
1140  !      Compute the calibration coefficients.
1150  !
1160  ! C.   Move the bed to each of the extremes
1170  !      in the X, Y, and Z directions using
1180  !      the override switches and have the
1190  !      user tell the computer when each of
1200  !      these boundaries are hit.  Enter each
1210  !      of these onto the disk file.
```

```
1220  !
1230  ! D.  Disk file "Motor_coef":
1240  !
1250  !      1.  x_zero, x_slope
1260  !      2.  y_zero, y_slope
1270  !      3.  z_zero, z_slope
1280  !      4.  x_min, x_max
1290  !      5.  y_min, y_max
1300  !      6.  z_min, z_max
1310  !
1320  BEEP
1330     PRINTER IS 1
1340     GCLEAR
1350     OUTPUT 2 USING "#,B";255,75
1360     !
1370  PRINT "I. POTENTIOMETER CALIBRATION:        "
1380  PRINT "                                     "
1390  PRINT "   NOTE: 1. Probe must be horizontal"
1400  PRINT "          2. Arm must be parallel to "
1410  PRINT "             the bed axis           "
1420  PRINT "          3. ALPHA = BETA           "
1430  PRINT "                                     "
1440  PRINT "  A. Using the override switches,   "
1450  PRINT "     position the probe volume at   "
1460  PRINT "     the outer wall of the tip of   "
1470  PRINT "     the nozzle.                    "
1480  PRINT "                                     "
1490  PRINT "  B. I will need the nozzle O.D. and"
1500  PRINT "     the milling machine positon.   "
1510  PRINT "                                     "
1520  PRINT "( Hit <cont>)                        "
1530  PAUSE
1540  BEEP 1500,.1
1550  INPUT "1. Nozzle O.D. (inches)?",Nozzle_od
1560  BEEP 2000,.1
1570  INPUT "2. X (in), (+ into flume)?",X_1
1580  BEEP 2500,.1
1590  INPUT "3. y (in), (+ along flume to the right)?",Y_1
1600  BEEP 3000,.1
1610  INPUT "4. z (in), (+ upward)?",Z_1
1620  !
1630  CALL Read_pot("X",Vx_1)
1640  CALL Read_pot("Y",Vy_1)
1650  CALL Read_pot("Z",Vz_1)
1660  !
1670  !
1680    CALL Clear_screen
1690  PRINT "C. Move the milling machine to a new"
1700  PRINT "   position in 3-D, by at least 5   "
1710  PRINT "   inches in each direction.        "
1720    BEEP 3200,.1
1730  INPUT "X, Y, Z in inches?",X_2,Y_2,Z_2
1740    CALL Read_pot("X",Vx_2)
1750    CALL Read_pot("Y",Vy_2)
1760    CALL Read_pot("Z",Vz_2)
1770  !
1780  ! C.  Calculate the calibration coefficients
1790  !
1800    X_zero=-((Nozzle_od/2)+Vx_1*((X_2-X_1)/(Vx_2-Vx_1)))
1810    X_slope=(X_2-X_1)/(Vx_2-Vx_1)
```

```
1820     !
1830     Y_zero=-Vy_1*(Y_2-Y_1)/(Vy_2-Vy_1)
1840     Y_slope=(Y_2-Y_1)/(Vy_2-Vy_1)
1850     !
1860     Z_zero=-Vz_1*(Z_2-Z_1)/(Vz_2-Vz_1)
1870     Z_slope=(Z_2-Z_1)/(Vz_2-Vz_1)
1880     !
1890   !
1900   ! D. Find the physical boundaries for each
1910   !    direction.
1920   !
1930     CALL Clear_screen
1940   PRINT "1. Move the milling machine to the  "
1950   PRINT "   minimum value of 'x'. Hit <cont>."
1960   PRINT "(Away from the flume, backwards)     "
1970   PAUSE
1980     CALL Read_pot("X",V)
1990       X_min=X_zero+X_slope*V
2000   PRINT "2. Move the milling machine to the  "
2010   PRINT "   maximum value of 'x'.  (Towards  "
2020   PRINT "   the flume).  Hit <cont>.         "
2030   PAUSE
2040     CALL Read_pot("X",V)
2050       X_max=X_zero+X_slope*V
2060   PRINT "3. Move the milling machine to the  "
2070   PRINT "   minimum value of 'y'. Hit <cont>."
2080   PRINT "(To the left along the flume)        "
2090   PAUSE
2100     CALL Read_pot("Y",V)
2110       Y_min=Y_zero+Y_slope*V
2120   PRINT "4. Move the milling machine to the  "
2130   PRINT "   maximum value of 'y'. Hit <cont>."
2140   PAUSE
2150     CALL Read_pot("Y",V)
2160       Y_max=Y_zero+Y_slope*V
2170   PRINT "5. Move the milling machine to the  "
2180   PRINT "   minimum value of 'z'. Hit <cont>."
2190   PRINT "(Downwards)"
2200   PAUSE
2210     CALL Read_pot("Z",V)
2220       Z_min=Z_zero+Z_slope*V
2230   PRINT "6. Move the milling machine to the  "
2240   PRINT "   maximum value of 'z'. Hit <cont>."
2250   PAUSE
2260     CALL Read_pot("Z",V)
2270       Z_max=Z_zero+Z_slope*V
2280   !
2290   ! E. Write out the file "Motor_coef".
2300   !
2310   ON ERROR GOTO Purge_file
2320 Reenter: CREATE BDAT Filename$,1
2330   ASSIGN @File TO Filename$
2340   !
2350     OUTPUT @File;X_zero,X_slope
2360     OUTPUT @File;Y_zero,Y_slope
2370     OUTPUT @File;Z_zero,Z_slope
2380     !
2390     OUTPUT @File;X_min,X_max
2400     OUTPUT @File;Y_min,Y_max
2410     OUTPUT @File;Z_min,Z_max
```

```
2420      !
2430      ASSIGN @File TO *
2440      SUBEXIT
2450 Purge_file:  PURGE Filename$
2460              GOTO Reenter
2470      !
2480  SUBEND
2490  !
2500  !
2510  !
2520  !
2530  !
2540  !
2550  SUB Read_pot(Direction$,Value)
2560  !
2570  !
2580  ! Read one potentiometer and return a volt-
2590  !   age.
2600  !
2610  !   R3 -- Pot X
2620  !   R4 -- Pot Y
2630  !   R5 -- Pot Z
2640  !
2650  !
2660  IF Directin$="X" THEN Relay=3
2670  IF Direction$="Y" THEN Relay=4
2680  IF Direction$="Z" THEN Relay=5
2690  !
2700  !
2710     OUTPUT 723;"OP,1,0T"
2720     OUTPUT 723;"OB,1,";Relay;",1T"
2730  !
2740  OUTPUT 723;"IP,3T"
2750  ENTER 72301;Value
2760  !
2770  !
2780  SUBEND
2790  !
2800  !
2810  !
2820  !
2830  !
2840  !
2850  SUB Clear_screen
2860  !  Clear the CRT.
2870  !
2880     OUTPUT 2 USING "#,B";255,75
2890     GCLEAR
2900  SUBEND
2910  !
2920  !
2930  !
2940  !
2950  !
2960  !
2970  SUB Motor(Direction$,Rotation$)
2980  !
2990  !  Turn on the motor in the requested direc-
3000  !    tion (x,y,z) with the requestion rota-
3010  !    tion (CW, CCW).
```

```
3020  !
3030    Dir$=Direction$
3040    Rot$=Rotation$
3050    IF Dir$="X" AND Rot$="CW" THEN Lbit=2
3060    IF Dir$="X" AND Rot$="CCW" THEN Lbit=1
3070    IF Dir$="Y" AND Rot$="CW" THEN Lbit=4
3080    IF Dir$="Y" AND Rot$="CCW" THEN Lbit=3
3090    IF Dir$="Z" AND Rot$="CCW" THEN Lbit=5
3100    IF Dir$="Z" AND Rot$="CW" THEN Lbit=6
3110    !
3120    !
3130    OUTPUT 723;"OP,7,";2^Lbit;"T"
3140  SUBEND
3150  !
3160  !
3170  !
3180  !
3190  !
3200  !
3210  SUB Motor_stop
3220  !
3230  !  Stop all motors!
3240  !
3250    OUTPUT 723;"OP,7,0T"
3260  SUBEND
3270  !
3280  !
3290  !
3300  !
3310  !
3320  !
3330  SUB Retrieve_coef(Coef(*),Filename$)
3340  !
3350    OPTION BASE 1
3360  !
3370  !  Retrieve the potentiometer calibration
3380  !     coefficients from a disk file called
3390  !     "Motor_coef".  Place these in an array.
3400  !
3410    ASSIGN @File TO Filename$
3420    !
3430    FOR I=1 TO 12 STEP 2
3440      ENTER @File;Coef(I),Coef(I+1)
3450    NEXT I
3460    !
3470    ASSIGN @File TO *
3480  SUBEND
3490  !
3500  !
3510  !
3520  !
3530  !
3540  !
3550  SUB Label_point(X,Y,Z,Symbol$)
3560  !
3570  !  Label a point on the Flume diagram
3580  !     using the symbol specified.
3590  !
3600     PRINT TABXY(1,13),"(X,Y,Z) inches = ";
3610     PRINT USING "DDD.DDD";X,Y,Z
```

```
3620      !
3630      !
3640      IF Symbol$="X" THEN
3650        !
3660        MOVE 14+Y,24+Z
3670          IMOVE -.2,-.2
3680          IDRAW 0,.4
3690          IDRAW .4,0
3700          IDRAW 0,-.4
3710          IDRAW -.4,0
3720        !
3730        MOVE 14+Y,11+X
3740          IMOVE -.2,-.2
3750          IDRAW 0,.4
3760          IDRAW .4,0
3770          IDRAW 0,-.4
3780          IDRAW -.4,0
3790      END IF
3800      !
3810      IF Symbol$="+" THEN
3820        !
3830        MOVE 14+Y,11+X
3840          IMOVE 0,.2
3850          IDRAW -.2,-.4
3860          IDRAW .4,0
3870          IDRAW -.2,.4
3880        !
3890        MOVE 14+Y,24+Z
3900          IMOVE 0,.2
3910          IDRAW -.2,-.4
3920          IDRAW .4,0
3930          IDRAW -.2,.4
3940      END IF
3950    SUBEND
3960    !
3970    !
3980    !
3990    !
4000    !
4010    !
4020    SUB Move_ldv_to(X,Y,Z,Length_arm,Length_probe,Angle_arm,Angle_probe,Coef(*
))
4030    DEG
4040    !
4050      OUTPUT 723;"CC,1T"
4060      OUTPUT 723;"SF,3,3,3,1.25,12T"
4070      PRINTER IS 1
4080      ON KEY 0 LABEL "ABORT" CALL Stop_all
4090      OPTION BASE 1
4100      Tolerance=.006
4110    !
4120    !  Move the probe to the position indicated
4130    !  in (inches) relative to the nozzle tip.
4140    !
4150    !  Dimensions are in inches and degrees
4160    !
4170    !  A.1  Load in the calibration coefficients.
4180    !
4190        X_zero=Coef(1)
4200        X_slope=Coef(2)
```

```
4210      Y_zero=Coef(3)
4220      Y_slope=Coef(4)
4230      Z_zero=Coef(5)
4240      Z_slope=Coef(6)
4250      X_min=Coef(7)
4260      X_max=Coef(8)
4270      Y_min=Coef(9)
4280      Y_max=Coef(10)
4290      Z_min=Coef(11)
4300      Z_max=Coef(12)
4310    !
4320    !  A.2 Move the probe
4330    !
4340    CALL Probe_moe(Angle_probe)
4350    !
4360    !  B.  Does (X,Y,Z) lie within the per-
4370    !        mitted boundaries?
4380    !
4390      IF X>X_max OR X<X_min OR Y>Y_max OR Y<Y_min OR Z>Z_max OR Z<Z_min THEN

4400        BEEP 1700,.5
4410        BEEP 2000,.5
4420        BEEP 1700,.5
4430        BEEP 2000,.5
4440        PRINT "Desired point is out of range!"
4450        SUBEXIT
4460      END IF
4470    !
4480    ! C.  Find out where the probe is now, draw
4490    !       the flume on the CRT, and label the
4500    !       desired position.
4510    !
4520    !  1.  Sound warning, movement immenent!
4530    !
4540       CALL Clear_screen
4550       PRINT TABXY(1,12),"MOVEMENT OF MILLING MACHINE IMMINENT!!!"
4560         FOR I=1 TO 4
4570           BEEP 1200,.1
4580           BEEP 1700,.1
4590           BEEP 2200,.1
4600           BEEP 2700,.1
4610         NEXT I
4620         CALL Clear_screen
4630       !
4640       !
4650       OUTPUT 723;"CC,1T"    !CLEAR RELAY CARD.
4660    CALL Position("X",X_actual,Valu_shaft,Length_arm,Length_probe,Angle_arm,
Angle_probe,Coef(*))
4670    CALL Position("Y",Y_actual,Valu_shaft,Length_arm,Length_probe,Angle_arm,
Angle_probe,Coef(*))
4680    CALL Position("Z",Z_actual,Valu_shaft,Length_arm,Length_probe,Angle_arm,
Angle_probe,Coef(*))
4690       !
4700       !
4710       CALL Draw_flume
4720       CALL Label_point(X_actual,Y_actual,Z_actual,"+")
4730       CALL Label_point(X,Y,Z,"X")
4740    !
4750    !  D.  Move each motor to bring the error
4760    !        between actual and desired position
```

```
4770  !      into tolerance.
4780  !
4790          X_old=X_actual
4800 X_node:  Xerror=X-X_actual
4810          IF ABS(Xerror)>Tolerance THEN
4820            IF Xerror>0 THEN Rot$="CCW"
4830            IF Xerror<0 THEN Rot$="CW"
4840              CALL Motor("X",Rot$)
4850    CALL Position("X",X_actual,Valu_shaft,Length_arm,Length_probe,Angle_arm,
Angle_probe,Coef(*))
4860              CALL Plot_path(X_old,Y_actual,Z_actual,X_actual,Y_actual,Z_actu
al)
4870                   X_old=X_actual
4880            GOTO X_node
4890          END IF
4900 !
4910 !
4920          Y_old=Y_actual
4930 Y_node:  Yerror=Y-Y_actual
4940          IF ABS(Yerror)>Tolerance THEN
4950            IF Yerror>0 THEN Rot$="CCW"
4960            IF Yerror<0 THEN Rot$="CW"
4970              CALL Motor("Y",Rot$)
4980    CALL Position("Y",Y_actual,Valu_shaft,Length_arm,Length_probe,Angle_arm
Angle_probe,Coef(*))
4990              CALL Plot_path(X_actual,Y_old,Z_actual,X_actual,Y_actual,Z_actu
al)
5000                   Y_old=Y_actual
5010            GOTO Y_node
5020          END IF
5030 !
5040 !
5050          Z_old=Z_actual
5060 Z_node:  Zerror=Z-Z_actual
5070          IF ABS(Zerror)>Tolerance THEN
5080            IF Zerror>0 THEN Rot$="CW"
5090            IF Zerror<0 THEN Rot$="CCW"
5100              CALL Motor("Z",Rot$)
5110    CALL Position("Z",Z_actual,Valu_shaft,Length_arm,Length_probe,Angle_arm,
Angle_probe,Coef(*))
5120              CALL Plot_path(X_actual,Y_actual,Z_old,X_actual,Y_actual,Z_actu
al)
5130              Z_old=Z_actual
5140            GOTO Z_node
5150          END IF
5160 !
5170 !
5180    CALL Motor_stop
5190 !
5200 !
5210    FOR I=1 TO 4
5220        BEEP 2400,.2
5230        BEEP 4800,.2
5240    NEXT I
5250 !
5260 SUBEND
5270 !
5280 !
5290 !
5300 !
```

```
5310 !
5320 !
5330 SUB Position(Direction$,Value,Valu_shaft,Length_arm,Length_probe,Angle_arm,
Angle_probe,Coef(*))
5340   !
5350   OPTION BASE 1
5360   DEG
5370   !
5380   !  Return the position (inches) for the
5390   !    appropriate direction relative to the
5400   !    nozzle tip.
5410   !
5420      CALL Read_pot(Direction$,Voltage)
5430   !
5440      X_zero=Coef(1)
5450      X_slope=Coef(2)
5460      Y_zero=Coef(3)
5470      Y_slope=Coef(4)
5480      Z_zero=Coef(5)
5490      Z_slope=Coef(6)
5500   !
5510   !
5520      Dir$=Direction$
5530      !
5540      IF Dir$="X" THEN
5550         Valu_shaft=X_zero+X_slope*Voltage
5560         Value=Valu_shaft-Length_arm*COS(Angle_arm)*TAN(45-Angle_arm/2.0)
5570      END IF
5580      !
5590      IF Dir$="Y" THEN
5600         Valu_shaft=Y_zero+Y_slope*Voltage
5610         Value=Valu_shaft+Length_probe*(1.-COS(Angle_probe))+Length_arm*COS(
Angle_arm)
5620      END IF
5630      !
5640      IF Dir$="Z" THEN
5650         Valu_shaft=Z_zero+Z_slope*Voltage
5660         Value=Valu_shaft-Length_probe*SIN(Angle_probe)
5670      END IF
5680      !
5690   SUBEND
5700   !
5710   !
5720   !
5730   !
5740   !
5750   !
5760   SUB Stop_all
5770   !  STOP ALL MOTORS AND QUIT
5780   !
5790     GCLEAR
5800     CALL Motor_stop
5810     PRINT "MOTOR CONTROL ABORTED!!!"
5820     PRINT "(HIT <CONT> TO CONTINUE)"
5830     PAUSE
5840   SUBEND
5850   !
5860   !
5870   !
5880   !
```

```
5890  !
5900  !
5910  SUB Plot_path(X1,Y1,Z1,X2,Y2,Z2)
5920  !
5930  !  Plot the path of the probe on the flume
5940  !    diagram as the motors move the bed.
5950  !
5960  ! Lower plot followed by upper plot.
5970  !
5980     PRINT TABXY(1,13),"(X,Y,Z) inches = ";
5990     PRINT USING "DD.DDD";X2,Y2,Z2
6000     MOVE 14+Y1,24+Z1
6010     DRAW 14+Y2,24+Z2
6020     !
6030     MOVE 14+Y1,11+X1
6040     DRAW 14+Y2,11+X2
6050     !
6060  SUBEND
```

4. T_SUBS

```
10      !    T_SUBS
20      !
30      !
40      SUB T_couple(Temperature,Choice$,Scale$,No_readings,Stdev)
50      !
60      !   SUBPROGRAM T_COUPLE
70      !
80      !   by Bill Culbreth
90      !   19 April 1984
100     !
110     !   PURPOSE:  This program is designed to
120     !      read type T or E thermocouples and
130     !      return the actual temperature to
140     !      the calling routine.
150     !
160     !   Temperature -- Temperature from the
170     !      thermocouple in degrees F or C from
180     !      the thermocouple identified by
190     !      "Choice$".
200     !   Choice$ -- Thermocouple choice, current-
210     !      ly:  "AMBIENT", "NOZZLE", or, "PROBE".
220     !   Scale$  -- "F" for Fahrenheit or "C" for
230     !      Celsius, or "H" for histogram in "C".
240     !   No_readings -- How many readings of the
250     !      same thermocouple should the routine
260     !      take?
270     !   Stdev -- The standard deviation of the
280     !      temperature for the indicated number
290     !      of readings in the units given by
300     !      "Scale$".
310     !
320     !
330     !   1.  Open all relays and initialize the
340     !       A/D converter.
350     !
360         OUTPUT 723;"CC,1T"
370         OUTPUT 723;"SF.3,3,3,1.25,12T"
380         !
390         OUTPUT 723;"OP,1,0T"
400     !
410     !   2.  Close the chosen relays.
420     !
430     !     a.  Ambient T -- Type T, relays 6,8.
440     !     b.  Nozzle  T -- Type T, relays 7,8.
450     !     c.  Probe   T -- Type E, relay  9.
460     !
470         IF Scale$="H" THEN
480            ! Plot a histogram using Celsius.
490            Histogram$="YES"
500            Scale$="C"
510         ELSE
520            Histogram$="NO"
530         END IF
540         !
550         !
560         IF Choice$="AMBIENT" THEN
570            Type$="T"
580            OUTPUT 723;"OB,1,6,1,8,1T"
590         END IF
600         !
```

```
610      IF Choice$="NOZZLE" THEN
620         Type$="T"
630         OUTPUT 723;"OB,1,7,1,8,1T"
640      END IF
650      !
660      IF Choice$="PROBE" THEN
670         Type$="E"
680         OUTPUT 723;"OB,1,9,1T"
690      END IF
700      !
710    !
720    !  3.  Take an A/D conversion and convert
730    !      into temperature.
740    !
750      Sum=0
760      Sum1=0
770    !
780      WAIT 1
790      IF Histogram$="YES" THEN GOSUB Set_up_histo
800      !
810      !
820      FOR I=1 TO No_readings
830         OUTPUT 723;"IP,3T"
840         ENTER 72301;A
850            IF Type$="T" THEN
860               A=A/1000
870            END IF
880            IF Type$="E" THEN
890               A=A/1000
900            END IF
910     !               PRINT "V(mV) = ";A
920         GOSUB Convert_t
930     !               BEEP A*100,.01
940           Sum=Sum+A
950           Sum1=Sum1+A*A
960           !
970           IF Histogram$="YES" THEN GOSUB Plot_point
980           !
990      NEXT I
1000   !
1010     Temperature=Sum/No_readings
1020     !
1030     !
1040     IF No_readings=1 THEN
1050        Stdev=0
1060     ELSE
1070        Stdev=SQR(ABS((Sum1-No_readings*Temperature^2)/(No_readings-1)))
1080     END IF
1090   !
1100   !  4.  Open all relays.
1110   !
1120     OUTPUT 723;"OP,1,0T"
1130   !
1140   SUBEXIT
1150   !
1160   !
1170   !
1180   Convert_t:   !
1190                ! This subroutine converts (mV)
1200                ! from a thermocouple into temp-
```

```
1210              ! erature.
1220              !
1230  !
1240  IF Type$="T" THEN
1250    A=2.5661297E+1*A-6.1954869E-1*A*A+2.2181644E-2*A^3-3.55009E-4*A^4
1260  END IF
1270  !
1280  !
1290  IF Type$="E" THEN
1300    A=1.7022525E+1*A-2.2097240E-1*A*A+5.4809314E-3*A^3-5.7669892E-5*A^4
1310  END IF
1320  !
1330  !  Fix the scale.
1340  !
1350     IF Scale$="F" THEN A=1.8*A+32
1360  RETURN
1370  !
1380  !
1390  !
1400  !
1410  !
1420  !
1430 Set_up_histo:  !
1440        !
1450        !  Set up a histogram of temperature
1460        !  versus number of counts.
1470        !
1480        !  1. Zero out the Height(*) array.
1490        !
1500        DIM Height(203)
1510        !
1520        FOR I=1 TO 202
1530          Height(I)=0
1540        NEXT I
1550        !
1560        !
1570        DUMP DEVICE IS 701
1580        GINIT
1590        OUTPUT 2 USING "#,B":255,75
1600        GRAPHICS ON
1610        FRAME
1620        WINDOW -100,100,-10,100
1630        MOVE -65,92
1640          CSIZE 7
1650          LABEL "TEMPERATURE HISTOGRAM"
1660          AXES 25,10,0,0,4,5,3
1670          PEN -1
1680          MOVE 0,-10
1690          DRAW 0,0
1700          MOVE 0,30
1710          DRAW 0,100
1720          PEN 1
1730       !
1740       !  Take 10 temperature readings to get
1750       !    the scale.
1760       !
1770          Sum=0
1780          Sum1=0
1790       FOR I=1 TO 10
1800          OUTPUT 723;"IP,DT"
```

```
1810          ENTER 72301;A
1820       NEXT I
1830       !
1840       FOR I=1 TO 100
1850          OUTPUT 723;"IP,3T"
1860          ENTER 72301;A
1870          IF Type$="T" THEN
1880             A=A/1000
1890          END IF
1900          IF Type$="E" THEN
1910             A=A/1000
1920          END IF
1930          GOSUB Convert_t
1940          Sum=Sum+A
1950          Sum1=Sum1+A*A
1960       NEXT I
1970          T_mean=Sum/100
1980          Sd=SQR(ABS(Sum1-100*T_mean^2)/99)
1990          !
2000          !  Change the window to extent from
2010          !      -4*Sd to +4*Sd.
2020          !
2030          MOVE -12,-7
2040          CSIZE 4
2050          LABEL USING "DDD.DD";T_mean
2060          MOVE 15,-7
2070          LABEL "C"
2080          !
2090          MOVE -87,-7
2100          LABEL USING "DDD.DD";T_mean-3*Sd
2110          MOVE 63,-7
2120          LABEL USING "DDD.DD";T_mean+3*Sd
2130          !
2140          !
2150          MOVE 5,47
2160          LABEL "50"
2170          !
2180          !
2190          !  3. Calculate the window temper-
2200          !     ature interval.
2210          !
2220          Interval=4*Sd/100
2230          !
2240          T_min=T_mean-4*Sd
2250          T_max=T_mean+4*Sd
2260          !
2270          WINDOW T_min,T_max,-10,100
2280    RETURN
2290    !
2300    !
2310    !
2320    !
2330    !
2340    !
2350 Plot_point:  !
2360          !
2370          ! Plot each temperature point as
2380          !    received on the histogram.
2390          !
2400          Freq=((A-T_min)/(T_max-T_min))*3000+1000
```

```
2410       BEEP Freq,.01
2420          J=0
2430          !
2440       FOR T=T_min TO T_max STEP Interval
2450          J=J+1
2460          IF A<=T THEN
2470             Height(J)=Height(J)+1
2480             GOTO Continue
2490          END IF
2500       NEXT T
2510       !
2520       !  Draw the point.
2530       !
2540 Continue:       !
2550          !
2560          !
2570          MOVE A,Height(J)
2580          CSIZE 4
2590          LABEL "."
2600       !
2610       !  Label the number of points and
2620       !    the temperature.
2630       !
2640          PRINT TABXY(2,3);"T(C)=";A
2650          PRINT TABXY(2,5);"Sample #";I
2660       !
2670  RETURN
2680  !
2690  !
2700  !
2710  !
2720  !
2730  !
2740  SUBEND
```

5. T_CAL

```
10   !    T_CAL
20   !
30   !This program records the output from the
40   !ambient, probe and nozzle themocouples
50   !for calibration purposes.  The data is
60   !printed as well as transmitted to a data
70   !file.
80   !
90   !Data is recorded in the following order:
100  !        a. Ambient temperature
110  !        b. Probe temperature
120  !        c. Nozzle temperature
130  !
140  !Place the ambient, probe and nozzle
150  !thermocouples into a bucket of warm water,
160  !execute this program, then add ice.
170  !
180  !
190  LOADSUB ALL FROM "T_SUBS"
200  !
210  !  Identify the BDAT file
220  !
230  INPUT "NAME OF FILE WHERE DATA IS TO BE STORED?",Filename1$
240  !
250  !  Identify the number of data points desired
260  !
270  INPUT "NUMBER OF DATA POINTS DESIRED?",Nitems
280  !
290  !  Identify the number of samples per
300  !  thermocouple per data point
310  !
320  INPUT "NUMBER OF SAMPLES PER DATA POINT?",Sample
330  !
340  !  Initialize a counter
350  !
360  I=1
370  !
380  !  Create a data file
390  !
400  Records=(Nitems*8*3/256)+2
410  CREATE BDAT Filename1$,Records
420  ASSIGN @File1 TO Filename1$
430  !
440  !  Take data and print results
450  !
460  CALL T_couple(T_ambient,"AMBIENT","C",Sample,Sd)
470  BEEP T*50,.05
480  PRINT "TAMB,SD=";T_ambient,Sd
490  CALL T_couple(T,"PROBE","C",Sample,St_dev)
500  BEEP T*50,.05
510  PRINT "TPROBE,SD=";T,St_dev
520  CALL T_couple(T_nozzle,"NOZZLE","C",Sample,Sd)
530  BEEP T*50,.05
540  PRINT "TNOZL,SD=";T_nozzle,Sd
550  !
560  !  Send data to the data file
570  !
580  OUTPUT @File1;T_ambient,T,T_nozzle
590  !
600  !  Test to see if finished
```

```
610  !
620  IF I=Nitems THEN
630     GOTO 760
640  ELSE
650     I=I+1
660     GOTO 460
670  END IF
680  !
690  !  Close the data file
700  !
710  OUTPUT @File1;-100
720  ASSIGN @File1 TO *
730  !
740  !  Alert the user...the job is completed
750  !
760  PRINT "ALL DONE"
770  BEEP
780  BEEP
790  END
```

6. PROBE_CAL

```
10   !   PROBE_CAL
20   !
30   !    This program will provide the data
40   !  necessary to calibrate probe deflection
50   !  if used in the following manner:
60   !
70   !    a. Switch OFF probe actuation power
80   !       at the probe base.
90   !
100  !    b. Swing the probe arm out of the
110  !       tank such that the probe is
120  !       accessable.
130  !
140  !    c. Attach the calibration panel to the
150  !       probe assembly, taking care not to
160  !       damage the glass probe.
170  !
180  !    d. Run the program...it will ask for
190  !       the desired position in mV.  The
200  !       following guidelines apply:
210  !
220  !            i. If it is desired to lower
230  !               the probe, type the extreme
240  !               value 9000 (anything >4600
250  !               will work, but 9000 is a
260  !               quick and easy number to
270  !               enter).
280  !
290  !           ii. If it is desired to raise
300  !               the probe, enter the extreme
310  !               value 30 (anything less than
320  !               940 will work).
330  !
340  !    e. The program will next ask which bit
350  !       is selected to be "high".  The
360  !       following guidelines apply:
370  !
380  !            i. If it is desired to lower the
390  !               probe, enter 7.
400  !
410  !           ii. If it is desired to raise the
420  !               probe, enter 8.
430  !
440  !    f. Switch ON probe activation power and
450  !       when the probe reaches a desired
460  !       degree of deflection, switch the
470  !       activation power OFF and record the
480  !       mV value printed on the screen.
490  !
500  !    g. Repeat steps (d) through (f) until
510  !       sufficient data is collected.  Load
520  !       this data into the program "POLYFIT"
530  !       and request a second order fit
540  !       (let X=angle and Y=mV).
550  !       Enter the coefficients derived by
560  !       this program into the appropriate
570  !       location in the program "PROBE_SUBS".
580  !
590  !
600  !   NOTE:  The relays will be activated on
```

```
610   !     digital low!  When the machine boots
620   !     up (hp-9826), all relay lines are
630   !     high (+5V).  The instructions below
640   !     will drop the voltage to zero.
650   !
660   INPUT "What is the desired position? (mV)",Voltage
670   PRINT "Desired position (mV) =";Voltage
680   !
690   !Read the actual motor position.
700   !  If the desired position is BELOW the
710   !  actual position, then tell the motor
720   !  to move UP.
730   !  If it is ABOVE, then tell it to go DOWN.
740   !
750   !
760   INPUT "WHICH BIT DO YOU WANT HIGH?",Lbit
770   !
780   !
790     OUTPUT 723;"OP,7,0T" !CLEAR ALL D/O
800     OUTPUT 723;"OP,7,";2^Lbit;"T"
810     !
820     OUTPUT 723;"CC,1T" !CLEAR A/D CARD
830     !
840   OUTPUT 723;"SF,3,3,3,1.25,12T" !FORMAT A/D
850   OUTPUT 723;"OB,1,10,1T" !CLOSE THE RELAY
860                           !THAT CONNECTS THE
870                           !A/D CARD TO THE
880                           !POTENTIOMETER
890   OUTPUT 723;"IP,3T" !START A/D CONVERSION
900   ENTER 72301;A      !ENTER A/D VALUE INTO A.
910   DISP A
920   BEEP ABS(A),.01
930   !
940   !  The following IF stops the motor when
950   !  it reaches it's limits.
960   !
970   IF (Lbit=7 AND A>4600) OR (Lbit=8 AND A<940) THEN
980      OUTPUT 723;"OP,7,0T"
990      BEEP
1000     PRINT "ALL DONE!!!"
1010  END IF
1020  GOTO 890
1030  END
```

7. MOTOR_CAL

```
10    !   MOTOR_CAL
20    !
30    !  PURPOSE:  Calibrate the potentiometers
40    !     used with the motors on the milling
50    !     machine.
60    !
70    !
80    OPTION BASE 1
90    DIM Coef(12)
100   !
110   OUTPUT 723;"SF,3,3,3,1.25,12T"
120   LOADSUB ALL FROM "MTR_SUBS"
130   LOADSUB ALL FROM "PROBE_SUBS"
140   File$="motor_coef"
150   !
160   CALL Calibrate(File$)
170   !
180   CALL Retrieve_coef(Coef(*),File$)
190     BEEP 2400,1
200   !
210   !
220   PRINT "What (x,y,z) position do you wish to"
230   PRINT "  move to? (inches relative to noz-"
240   PRINT "  zle)"
250   INPUT X,Y,Z
260   !
270   CALL Move_ldv_to(X,Y,Z,Length_arm,Length_probe,Angle_arm,Probe_angle,Coef(
*))
280   !
290   BEEP
300   GOTO 280
310   END
```

8. LOAD_XYZ

```
10      !   LOAD_XYZ
20      !
30      !   by Bill Culbreth
40      !
50      !   30 April 1984
60      !
70      !
80      !   PURPOSE:  This program will allow the
90      !      user to enter desired (x,y,z) posi-
100     !      tions of the milling machine relative
110     !      to the nozzle.  The values will be
120     !      stored on disk to be utilized later
130     !      by MAIN_T.
140     !
150     DIM X(500),Y(500),Z(500)
160     !
170     !   1.  Input the file name.
180     !
190         GOSUB Clear_screen
200     !
210         PRINTER IS 1
220         PRINT "Input the file name."
230         PRINT "( I suggest 'RUNXX' where 'XX'"
240         PRINT "  is the run number.          )"
250         PRINT
260         INPUT Filename$
270         !
280     !
290     !   2.  Begin inputting data.
300     !
310     !
320         GOSUB Clear_screen
330         !
340         PRINT "Do you wish to append a previous data file?"
350         INPUT Answer$
360         !
370         IF Answer$="YES" THEN
380            INPUT "Previous file name?",Old_file$
390            ASSIGN @File1 TO Old_file$
400            !
410            I=0
420 Loop1: !
430            I=I+1
440            ENTER @File1;X(I),Y(I),Z(I)
450            OUTPUT 701;"I,X,Y,Z=";I,X(I),Y(I),Z(I)
460            IF X(I)<>-100 THEN GOTO Loop1
470            ASSIGN @File1 TO *
480            PURGE Old_file$
490            Count=I-1
500         ELSE
510            Count=0
520         END IF
530            !
540         GOSUB Clear_screen
550         BEEP
560         !
570         PRINT "1. Input the desired position in "
580         PRINT "   inches as X, Y, and Z."
590         PRINT
600         PRINT "2. Terminate input by entering -100 "
```

```
610       PRINT "2. Terminate input by entering '-100'"
620       PRINT "    for X, Y, and Z."
630       PRINT "3. If you wish to enter an orientation angle."
640       PRINT "    enter '-999,orientation,0'."
650       PRINT
660       !
670 Begin: !
680         Count=Count+1
690         PRINT "Item #";Count
700         INPUT "(X,Y,Z) in inches?",X(Count),Y(Count),Z(Count)
710         OUTPUT 701;"I,X,Y,Z=";Count,X(Count),Y(Count),Z(Count)
720         IF X(Count)<>-100 THEN GOTO Begin
730       !
740         End_count=Count-1
750         !
760         !  All data points have been entered.
770         !
780         !  a.  Set up new softkeys.
790         !  b.  Explain softkeys.
800         !
810     GOSUB Clear_screen
820     !
830     PRINT "SOFTKEY LABELS:"
840     PRINT "  0 -- WRITE data to disk."
850     PRINT "  2 -- EDIT out bad data."
860     PRINT "  4 -- HARD copy the data on printer."
870     PRINT "  6 -- LIST data on the CRT."
880     PRINT "  8 -- STOP terminates the program."
890     !
900     ON KEY 0 LABEL "WRITE" GOSUB Write_data
910     ON KEY 2 LABEL "EDIT" GOSUB Edit_data
920     ON KEY 4 LABEL "HARD" GOSUB Hard_copy
930     ON KEY 6 LABEL "LIST" GOSUB List_data
940     ON KEY 8 LABEL "STOP" GOTO Terminate
950     GOTO 900
960     !
970     !
980     !
990     !
1000    !
1010    !
1020    !
1030 Clear_screen:   !
1040        !
1050        !  Clear the CRT display.
1060        !
1070        OUTPUT 2 USING "#,B";255,75
1080        GCLEAR
1090 RETURN
1100    !
1110    !
1120    !
1130    !
1140    !
1150    !
1160 Hard_copy: !
1170        !
1180        ! Print out all data to the printer.
1190        !
1200        PRINTER IS 701
```

```
1210        !
1220          PRINT "  Count      X(inches)      Y(count)      Z(count)"
1230          PRINT "  _____      _________      ________      ________"
1240          PRINT
1250        FOR I=1 TO End_count
1260          PRINT I,X(I),Y(I),Z(I)
1270        NEXT I
1280        !
1290        PRINTER IS 1
1300 RETURN
1310 !
1320 !
1330 !
1340 !
1350 !
1360 !
1370 List_data: !
1380     !
1390     ! List data to the CRT.
1400     !
1410     GOSUB Clear_screen
1420     !
1430     PRINT "There are";End_count;" data points."
1440     PRINT
1450     INPUT "Which point do I start with?",Start
1460     INPUT "which point do I end with?",End_data
1470     !
1480     FOR I=Start TO End_data
1490        PRINT "I,X,Y,Z=";I,X(I),Y(I),Z(I)
1500     NEXT I
1510  RETURN
1520  !
1530  !
1540  !
1550  !
1560  !
1570  !
1580 Edit_data:  !
1590        !
1600        !  Edit data.
1610        !
1620        GOSUB Clear_screen
1630        !
1640        INPUT "Which data point do you wish to alter?",I
1650        !
1660        PRINT
1670        PRINT "For point #";I;", (X,Y,Z) where:"
1680        PRINT X(I);",";Y(I);",";Z(I)
1690        !
1700        INPUT "Type in the new values:",X(I),Y(I),Z(I)
1710  RETURN
1720  !
1730  !
1740  !
1750  !
1760  !
1770  !
1780 Write_data:  !
1790        !
1800        !  Write data out to a file on disk.
```

```
1810         !
1820         X(End_count+1)=-100
1830         Y(End_count+1)=-100
1840         Z(End_count+1)=-100
1850         !
1860         Max_data=3*(End_count+1)
1870         !
1880         File_size=INT(Max_data*8/256)+1
1890         !
1900         CREATE BDAT Filename$,File_size
1910         ASSIGN @File TO Filename$
1920         !
1930         FOR I=1 TO End_count+1
1940           OUTPUT @File;X(I),Y(I),Z(I)
1950         NEXT I
1960         !
1970         GOSUB Clear_screen
1980         BEEP 2400,.3
1990         PRINT "File ";Filename$;" has been stored!"
2000         ASSIGN @File TO *
2010   RETURN
2020   !
2030   !
2040   !
2050   !
2060   !
2070   !
2080  Terminate:   !
2090         GOSUB Clear_screen
2100         PRINT "NORMAL TERMINATION OF PROGRAM!"
2110   END
```

9. SEND_DATA

```
10    !     SEND_DATA
20    !
30    !     To VAX,IBM,TRS-80.
40    !
50    !  HP-9826 TERMINAL PROGRAM
60    !  (REQUIRES BINARY ENHANCEMENT PROGRAM
70    !   "BEB"!)
80    !
90    !  JUNE 30, 1982
100   !  updated 1/5/83
110   !  updated 1/16/84
120   !
130   !  BILL CULBRETH
140   !
150   !
160   Sc=9     ! RS-232 IS SELECT CODE 9.
170   PRINTER IS 1   ! PRINTER IS CRT.
180   Pr=1           ! DEFAULT PRINTER IS CRT
190   Printer_choice=701 ! MY PRINTER IS 701.
200   Bits=7         ! BITS PER CHARACTER
210   Duplex=0       ! FULL DUPLEX
220   Baud=300       ! BAUD RATE
230   Computer=1     ! ASSUME IBM COMPUTER
240   !
250   OUTPUT Pr;"(300 BAUD, IBM assumed."
260   OUTPUT Pr;" Load the binary program BEB first"
270   OUTPUT Pr;" unless you have BASIC 2.0"
280   OUTPUT Pr;" SET MODEM ON <FULL DUPLEX>  "
290   OUTPUT Pr;" "
300   !
310   DIM Name$[200],Hp_file$[30],Aa(1500),Nump$[30]
320   INTEGER Isend
330   !
340   CONTROL Sc,3;Baud
350   CONTROL Sc,4;Bits-5+4 ! BITS/CHAR & #STOP BITS.
360   !
370   !
380   To_disk=0
390   Datadump=0
400   I_data=1
410   I=1
420   J=1
430   K=1
440   L=1
450   !
460   ON ERROR GOTO Errors
470   ON KEY 0 LABEL "Line Mode" GOTO Line_mode
480   ON KEY 5 LABEL "Terminal" GOTO Terminal
490   ON KEY 6 LABEL "To Crt" GOTO Pr_crt
500   ON KEY 7 LABEL "To Prt" GOTO Pr_prt
510   ON KEY 8 LABEL "DATA" GOTO Data_dump
520   !
530   !
540 Line_mode:     !
550        OUTPUT Pr;"(LINE RECEPTION MODE)"
560 Begin: STATUS Sc,10;Y ! CHECK FOR FULL BUFFER
570•
580        IF BIT(Y,0)=0 THEN GOTO Begin
590   !
600   !  RECEIVE ROUTINE.
```

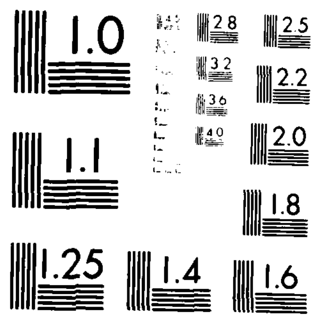
AD-A152 224 DESIGN AND CONSTRUCTION OF A COMPUTER CONTROLLED MIROTHERMOCOUPLE PROBE FOR THE STUDY OF BUOYANT JETS (U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA R J MATOUSHEK SEP 84 2/2

UNCLASSIFIED F/G 9/2 NL

END

FILMED

DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
610   !
620 Receive:     STATUS Sc,6;A
630       B=A
640       OUTPUT Pr USING "#,A";CHR$(B)
650       IF B=63 AND Datadump=1 THEN GOTO Data_dump
660       IF B=13 AND Computer=3 THEN OUTPUT Pr;CHR$(13)
670       GOTO Begin
680   !
690   !   TRANSMIT ROUTINE.
700   !
710*
720               IF Duplex=0 THEN
730                   IF NUM(Key$)<>255 THEN OUTPUT Pr USING "#,A";Key$
740                   IF NUM(Key$)=255 THEN OUTPUT Pr;" "
750               END IF
760               IF Computer=1 AND NUM(Key$)=8 THEN Key$=CHR$(64)
770               !
780               !   the previous line gives an @
790               !   for a backspace for the IBM.
800               !
810               IF Computer=5 AND NUM(Key$)=8 THEN Key$=CHR$(127)
820               !
830               !   THE VAX/VMS REQUIRES A DELETE
840               !   SYMBOL FOR A BACKSPACE.
850               !
860               IF NUM(Key$)=255 THEN Key$=CHR$(13)
870               OUTPUT Sc USING "#,A";Key$
880               GOTO Begin
890   !
900   !
910   !
920   !   DATA FILE OUT TO THE HOST COMPUTER.
930   !
940   !
950   !
960 Data_dump:    !
970               IF I_data=1 THEN GOSUB Open_file
980               !
990               IF Datadump=0 THEN GOTO Begin
1000                 IF Computer=1 THEN WAIT .3
1010                 ! wait for the slow IBM.
1020                 BEEP 1000+RND*1500,.05
1030                 OUTPUT Pr;"A(";I_data;")=";
1040                 OUTPUT Pr;Aa(I_data)
1050                 GOSUB Send_number
1060              IF Aa(I_data)=-200 THEN
1070                    I_data=1
1080                    Datadump=0
1090              END IF
1100              I_data=I_data+1
1110 GOTO Begin
1120 !
1130 !
1140 !   ERROR HANDLING SUBROUTINE
1150 !
1160 Errors:   OFF ERROR
1170           Close_file=-200
1180           !   FIRST, END OF FILE ERROR.
1190           IF ERRN=59 THEN
1200              Aa(I)=-200
```

```
1210            GOTO 2000 ! RETURN AFTER ERROR.
1220          END IF
1230          !
1240          IF ERRN<>59 THEN OUTPUT Pr;"<error #";ERRN;" generated.>"
1250          IF ERRN=54 THEN OUTPUT Pr;"(FILE <";Hp_file$;"> ALREADY THERE!"
1260          IF ERRN=54 THEN GOTO Created
1270          IF ERRN=56 THEN OUTPUT Pr;"<FILE (";Hp_file$;" IS NOT ON DISK.>"
1280          ASSIGN @File TO *
1290 GOTO Line_mode
1300 !
1310 !
1320 !  OUTPUT TO CRT.
1330 !
1340 Pr_crt:  Pr=1
1350          GOTO Line_mode
1360 !
1370 !
1380 !  OUTPUT TO PRINTER.
1390 !
1400 Pr_prt:  Pr=Printer_choice
1410 GOTO Line_mode
1420 !
1430 !
1440 !  CHANGE THE TERMINAL CHARACTERISTICS.
1450 !
1460 Terminal:  !
1470            OUTPUT Pr;"     1.  Baud Rate =";Baud
1480            OUTPUT Pr;"     2.  Bits/Char =";Bits
1490            OUTPUT Pr;"     3.  Duplex    =";Duplex
1500            OUTPUT Pr;"         [1=full,0=half]"
1510            OUTPUT Pr;"     4.  Computer  =";Computer
1520            OUTPUT Pr;"         [IBM=1, VAX/UNIX=2, "
1530            OUTPUT Pr;"          TRS-80=3, Cyber=4, vax/vms=5]"
1540            OUTPUT Pr;" "
1550            INPUT "Change which one?",Which
1560            IF Which=1 THEN INPUT "To?",Baud
1570            IF Which=2 THEN INPUT "To?",Bits
1580            IF Which=3 THEN INPUT "To?",Duplex
1590            IF Which=4 THEN INPUT "To?",Computer
1600            IF Computer=1 THEN Duplex=0
1610            IF Computer=3 THEN Duplex=0
1620            IF Computer=3 THEN Bits=8
1630            IF Computer=5 THEN Duplex=1
1640 GOTO Line_mode
1650 !
1660 !
1670 !
1680 Open_file:  !
1690             !  Open a file to read data from
1700             !  disk.
1710             !
1720      Datadump=1
1730      !
1740        INPUT "Is this LDV data? (1=YES)",Ldv$
1750        IF Ldv$="1" THEN
1760          INPUT "Experiment #?",Experiment$
1770        ELSE
1780          OUTPUT Pr;"Data file out of HP to host."
1790          INPUT "File name?",Hp_file$
1800        END IF
```

```
1810        !
1820        IF Ldv$="1" THEN
1830        Hp_file$=Experiment$&"_RESULT"
1840      END IF
1850      !
1860      !  Read the file off of disk.
1870      !
1880      ASSIGN @File TO Hp_file$
1890      I=1
1900      Check=0
1910      BEEP
1920      BEEP
1930      OUTPUT Pr;"(Working on file <";Hp_file$;">.)"
1940      !
1950*
1960        ENTER @File;Aa(I)
1970        Check=Aa(I)
1980        I=I+1
1990*
2000      !
2010      ASSIGN @File TO *
2020        Datadump=1
2030 RETURN
2040 !
2050 !
2060 !
2070 Send_number: !
2080                 ! SEND A NUMBER ONE CHARACTER AT
2090                 ! A TIME TO THE HOST COMPUTER.
2100                 !
2110           Numb$=VAL$(Aa(I_data))
2120           Length=LEN(Numb$)
2130           !
2140           IF ((Ldv$="1") AND (I_data>13)) THEN
2150             Posit=POS(Numb$,".")
2160             IF (Posit<>0) THEN Length=Posit+2
2170           END IF
2180           !
2190           FOR I=1 TO Length
2200             Numeric=NUM(Numb$[I,I])
2210             OUTPUT Sc USING "#,A";Numb$[I,I]
2220           NEXT I
2230           !
2240           OUTPUT Sc USING "#,A";CHR$(13)
2250 RETURN
2260 !
2270 !
2280 !
2290 END
```

APPENDIX C

MAINFRAME PROGRAMS

1. TCAL

```
C
C
C     TCAL
C
C     PURPOSE:  THIS PROGRAM PLACES THERMOCOUPLE CALIBRATION DATA
C               RECEIVED FROM THE HP-9826 MICROCOMPUTER INTO A
C               MORE WORKABLE FORMAT FOR USE IN THE PROGRAM TFIT.
C
C
      DIMENSION TPROBE(200),TAMB(200),TNOZZ(200)
      I=1
   10 READ(07,*) TAMB(I)
      IF(TAMB(I).EQ.-200.0) GO TO 20
      READ(07,*) TPROBE(I)
      READ(07,*) TNOZZ(I)
      NITEMS=I
      I=I+1
      GO TO 10
   20 CONTINUE
      WRITE(6,40)
      DO 30 I=1,NITEMS
         WRITE(6,50)I,TAMB(I),TPROBE(I),TNOZZ(I)
   30 CONTINUE
   40 FORMAT(8X,'AMBIENT (C)',2X,'PROBE (C)',2X,'NOZZLE (C)')
   50 FORMAT(1X,I3,6X,2(F6.3,6X))
      STOP
      END
```

2. TFIT

```
C
C     TFIT
C
C     PURPOSE:  THIS PROGRAM PERFORMS A FIRST ORDER CURVE FIT BY THE
C               LEAST SQUARES METHOD FOR TWO COLUMNS OF DATA.  THE
C               FIRST COLUMN LISTS VALUES OF X AND THE SECOND LISTS
C               VALUES OF F(X) OR Y.  FOR THERMOCOUPLE CALIBRATION,
C               LET X = THE AMBIENT TEMPERATURE, AND LET Y = EITHER
C               THE PROBE OR NOZZLE TEMPERATURE AS DESIRED.
C
      DIMENSION X(150),Y(150),XSQ(150),XY(150),YEST(150)
      SUMX=0.0
      SUMXSQ=0.0
      SUMXY=0.0
      SUMY=0.0
      SUMNUM=0.0
      SUMDEN=0.0
      READ(7,*)NITEMS
      DO 10 I=1,NITEMS
         READ(7,*)X(I),Y(I)
         XSQ(I)=X(I)**2
         XY(I)=X(I)*Y(I)
         SUMX=SUMX+X(I)
         SUMXSQ=SUMXSQ+XSQ(I)
         SUMXY=SUMXY+XY(I)
         SUMY=SUMY+Y(I)
   10 CONTINUE
      XITEMS=FLOAT(NITEMS)
      A=(SUMY*SUMXSQ-SUMX*SUMXY)/(XITEMS*SUMXSQ-SUMX**2)
      B=(XITEMS*SUMXY-SUMX*SUMY)/(XITEMS*SUMXSQ-SUMX**2)
      YBAR=SUMY/XITEMS
      DO 20 I=1,NITEMS
         YEST(I)=B*X(I)+A
      WRITE(6,*)YEST(I), Y(I)
         SUMNUM=SUMNUM+(YEST(I)-YBAR)**2
         SUMDEN=SUMDEN+(Y(I)-YBAR)**2
   20 CONTINUE
      RSQ=SUMNUM/SUMDEN
      WRITE(6,*)A,B,RSQ
      STOP
      END
```

3. JETCURV

```
C
C     JETCURV
C
C
C     PURPOSE:   THIS PROGRAM FITS JET TRAJECTORY DATA TO THE EQUATION
C                Z=A*Y/(B+Y) BY THE LEAST SQUARES METHOD AND COMPUTES A
C                CORRELATION COEFFICIENT (R**2).  DATA POINTS ARE READ
C                IN FREE FORMAT FROM UNIT "7" DISK.
C
C
      DIMENSION Y(50),Z(50),YY(50),ZZ(50),YSQ(50),YZ(50),ZEST(50)
      SUMY=0.0
      SUMYSQ=0.0
      SUMYZ=0.0
      SUMZ=0.0
      SUMNUM=0.0
      SUMDEN=0.0
      READ(7,*)NITEMS,FACTOR
      DO 10 I=1,NITEMS
         READ(7,*)Y(I),Z(I)
         Y(I)=Y(I)/FACTOR
         YY(I)=1.0/Y(I)
         ZZ(I)=1.0/(Z(I)/FACTOR)
         YSQ(I)=YY(I)**2
         YZ(I)=YY(I)*ZZ(I)
         SUMY=SUMY+YY(I)
         SUMYSQ=SUMYSQ+YSQ(I)
         SUMYZ=SUMYZ+YZ(I)
         SUMZ=SUMZ+ZZ(I)
   10 CONTINUE
      XITEMS=FLOAT(NITEMS)
      AA=(SUMZ*SUMYSQ-SUMY*SUMYZ)/(XITEMS*SUMYSQ-SUMY**2)
      BB=(XITEMS*SUMYZ-SUMY*SUMZ)/(XITEMS*SUMYSQ-SUMY**2)
      A=1.0/AA
      B=A*BB
      ZBAR=SUMZ/XITEMS
      DO 20 I=1,NITEMS
         ZEST(I)=BB*YY(I)+AA
         WRITE(6,30)ZZ(I),ZEST(I)
         SUMNUM=SUMNUM+(ZEST(I)-ZBAR)**2
         SUMDEN=SUMDEN+(ZZ(I)-ZBAR)**2
   20 CONTINUE
      RSQ=SUMNUM/SUMDEN
      WRITE(6,40)A,B,RSQ
   30 FORMAT(1X,'Z (ACTUAL) =',F12.8,3X,'Z (EST) =',F12.8)
   40 FORMAT(1X,'A= ',F12.8,2X,'B= ',F12.8,2X,'R**2= ',F12.8)
      STOP
      END
```

4. GRAB

```
C
C       GRAB
C
C       PURPOSE:  DATA TRANSFER FROM THE HP-9826MICROCOMPUTOR
C       TO THE IBM.
C
C       BY BILL CULBRETH
C       FOR ME2410, FALL QUARTER, 1982
C
C       FILEDEF 05 TERMINAL
C       FILEDEF 06 TERMINAL
C       FILEDEF 07 DISK MYDATA DATA (PERM)
C
C       GLOBAL TXTLIB FORTMOD2 MOD2EEH
C
C            TYPE IN THE ABOVE 4 LINES TO MAKE THIS
C                 FORTRAN PROGRAM RUN.
C
C
        DIMENSION DATA(3000)
        I=1
        WRITE(6,80)
80      FORMAT(2X,'BEGIN INPUTING DATA FROM THE HP-9826')
C
C
10      CONTINUE
        READ(5,*) DATA(I)
        I=I+1
        IF(DATA(I-1).NE.-200) GOTO 10
C
        NITEMS = I-1
6       FORMAT(2X,I5,'  DATA POINTS WERE ENTERED.')
        WRITE(6,6) NITEMS
C
C       NOW THAT ALL DATA HAS BEEN ENTERED, WRITE IT OUT ON
C       DISK.
C
5       FORMAT(2X,'DATA(',I5,') = ',1F15.5)
        I=1
20      WRITE(7,*) DATA(I)
        I=I+1
        IF(DATA(I-1).NE.-200) GOTO 20
C
C            ALL DATA HAS BEEN WRITTEN ONTO DISK.
C
        STOP
        END
```

5. TDATA

```
C
C
C     TDATA
C
C     PURPOSE:  THIS PROGRAM PLACES THE BUOYANT JET DATA RECEIVED
C               FROM THE HP-9826 MICROCOMPUTER INTO A MORE ORDERLY
C               FORMAT.  IT ALSO CONVERTS UNITS OF LENGTH FROM SI
C               TO METRIC.
C
      DIMENSION X(200),Y(200),Z(200),TPROBE(200),TAMB(200),TNOZZ(200),
     1STDEV(200),XMM(200),YMM(200),ZMM(200)
      I=1
   10 READ(07,*) X(I)
      IF(X(I).EQ.-200.0) GO TO 20
      XMM(I)=25.4*X(I)
      READ(07,*) Y(I)
      YMM(I)=25.4*Y(I)
      READ(07,*) Z(I)
      ZMM(I)=25.4*Z(I)
      READ(07,*) TPROBE(I)
      READ(07,*) TAMB(I)
      READ(07,*) TNOZZ(I)
      READ(07,*) STDEV(I)
      NITEMS=I
      I=I+1
      GO TO 10
   20 CONTINUE
      WRITE(6,40)
      DO 30 I=1,NITEMS
        WRITE(6,50)I,XMM(I),YMM(I),ZMM(I),TPROBE(I),TAMB(I),TNOZZ(I),
     1STDEV(I)
   30 CONTINUE
   40 FORMAT(8X,'X (MM)',4X,'Y (MM)',4X,'Z (MM)',2X,'PROBE (C)',1X,
     1'AMBIENT (C)',1X,'NOZZLE (C)',1X,'STD DEV (C)')
   50 FORMAT(1X,I3,3X,2(F7.3,3X),F7.3,4X,4(F6.3,4X))
      STOP
      END
```

6. CONTOUR4

```
C
C        CONTOUR4
C
C
C        PURPOSE:  THIS PROGRAM IS DESIGNED TO DISPLAY BUOYANT JET
C                  TEMPERATURE DATA USING A CONTOUR PLOTTING PACKAGE
C                  AVAILABLE WITH DISSPLA.  THE FOLLOWING RAW DATA IS
C                  READ FROM A DISK FILE: PROBE POSITIONS IN XYZ
C                  COORDINATES RELATIVE TO THE NOZZLE TIP, AMBIENT
C                  TEMPERATURE, PROBE TEMPERATURE AND NOZZLE TEMPERATURE.
C                  THE PROBE AND NOZZLE TEMPERATURES ARE TRANSFORMED
C                  BY CALIBRATION COEFFICIENTS AND NORMALIZED WRT AMBIENT
C                  TEMPERATURE.  THE XYZ COORDINATES ARE CONVERTED TO
C                  XSW COORDINATES RELATIVE TO THE INTERSECTION OF THE
C                  DATA PLANE WITH THE CENTERLINE TRAJECTORY OF THE JET
C                  (THE S-AXIS IS TANGENT TO THE TRAJECTORY).  THIS
C                  PROGRAM ALSO COMPUTES THE RATE OF HEAT TRANSFER FROM
C                  THE JET TO THE AMBIENT.
C
C                  NOTE:  THE FOLLOWING VALUES MUST BE INSERTED AS THE
C                  FIRST LINE OF DATA IN FREE FORMAT: THE TOTAL NUMBER
C                  OF DATA POINTS (NITEMS), THE ACUTE ANGLE BETWEEN
C                  THE DATA PLANE AND HORIZONTAL (THETA) AND THE VERTICAL
C                  DISTANCE BETWEEN THE Z-AXIS AND THE INTERSECTION
C                  POINT DISCUSSED ABOVE (ZA), THE CENTERLINE VELOCITY
C                  IN M/S (VEL) AND JET WIDTH IN MM (WIDTH).
C
C
      DIMENSION X(100),Y(100),T(100),TP(100),TA(100),TN(100),TMAT(10,10)
      COMMON WORK(16000)
C
C
C* * * * READ DATA FROM UNIT "7" DISK * * * *
C
C
      READ(7,*)  NITEMS,THETA,ZA,VEL,WIDTH
C
      DO 20 I=1,NITEMS
         READ(7,*)  A,X(I),A,Y(I),TP(I),TA(I),TN(I),A
            Y(I) = (Y(I)-ZA)/SIN(THETA)
C
C                             CALIBRATION COEFFICIENTS
            TP(I)=.984809875*TP(I)+1.74079605
            TN(I)=.933223426*TN(I)+1.84754086
C
            T(I) = (TP(I) - TA(I))/(TN(I) - TA(I))
   20 CONTINUE
C
C  30 FORMAT(2X,'I,X,Y,T=',I5,3F15.5)
C
C
C* * * * DIMENSION OF TMAT * * * *
C
C
      IXDIM = 10
      IYDIM = 10
C
C
C
C* * * * FIND THE MAXIMUM AND MINIMUM TEMPERATURES * * * *
C
C
      TMIN = 100.0
      TMAX = 0.0
      XMAX=0.0
      YMAX=0.0
      XMIN=100.0
      YMIN=100.0
C
      DO 40 I=1,NITEMS
         IF(T(I) .GT. TMAX)   TMAX = T(I)
         IF(T(I) .LT. TMIN)   TMIN = T(I)
```

```
          IF(X(I).GT.XMAX)XMAX=X(I)
          IF(Y(I).GT.YMAX)YMAX=Y(I)
          IF(X(I).LT.XMIN)XMIN=X(I)
          IF(Y(I).LT.YMIN)YMIN=Y(I)
   40  CONTINUE
C
C* * * * DETERMINE THE MAXIMUM VALUES OF X AND Y WITHIN A
C         QUADRANT, ASSUMING THE STREAMWISE AXIS IS CENTERED
C         IN A SQUARE MATRIX PLANE
C
       DX=((XMAX-XMIN)/FLOAT(IXDIM))/1000.
       DY=((YMAX-YMIN)/FLOAT(IYDIM))/1000.
       QUADX=(FLOAT(IXDIM)/2.0)*DX
       QUADY=(FLOAT(IYDIM)/2.0)*DY
C
C* * * * FIND THE INCREMENTAL AREA FOR HEAT TRANSFER CALCULATIONS * * *
C
       AREA=DX*DY
C
C* * * * ESTABLISH THE INCREMENT SIZE FOR CONTOUR PLOTS * * * *
C
       TINCR=(TMAX-TMIN)/5.25
C
C
C* * * * GENERATE THE GRID * * * *
C
C
       CALL COMPRS
C      CALL TEK618
       CALL PAGE( 8.,9.)
       CALL BLOWUP (0.75)
       CALL PHYSOR(1.25,1.)
       CALL AREA2D(5.5,5.5)
C
       CALL HEIGHT(0.100)
       CALL CARTOG
       CALL INTAXS
       CALL XNAME('X (MM)$',100)
       CALL YNAME('W (MM)$',100)
              CALL XTICKS(2)
              CALL YTICKS(2)
              CALL YAXANG(90.0)
C
C
C
       CALL SWISSM
       CALL HEADIN('TEMPERATURE CONTOURS IN A BUOYANT JET$',100,2.,4)
       CALL HEADIN('WITH A CROSSFLOWING AMBIENT$',100,2.,4)
       CALL HEADIN('(25% FLOW RATE)$',100,1.,4)
       CALL HEADIN('(PLANE A)$',100,1.,4)
       CALL GRAF(-30.0,05.0,30.0,-30.0,05.0,30.0)
C
C* * * * GENERATE THE INTERPOLATED TEMPERATURE MATRIX "TMAT" * * *
C
       CALL BCOMON(16000)
       CALL ZBASE(TMIN)
           CALL BGNMAT(IXDIM,IYDIM)
           CALL GETMAT(X,Y,T,NITEMS,0)
           CALL ENDMAT(TMAT,0)
C
C* * * * SOLVE THE RATE OF HEAT TRANSFER * * * *
C
           QSUM=0.0
           DO 70 I=1,IXDIM
              DO 50 J=1,IYDIM
                 RO=(1.000433 + 3.51910E-5*TMAT(I,J)
     1           -5.5F2007E-6*TMAT(I,J)**2
     1           + 3.4129629E-8*TMAT(I,J)**3
     1           -?.740676E-11*TMAT(I,J)**4)*1000.
                 XX=QUADX-FLOAT(J-1)*DX
                 YY=QUADY-FLOAT(I-1)*DY
                 RADIUS=SQRT(XX**2 + YY**2)
```

```
                  V=VEL*EXP(-RADIUS**2/WIDTH**2)
                  Q=RO*AREA*V*.345*(TMAT(I,J)+273.16)
                  QSUM=QSUM+Q
   60          CONTINUE
   70       CONTINUE
            WRITE(6,80)QSUM
   80       FORMAT(1X,'Q= ',F20.5)
C
C* * * * PLOT THE CONTOURS * * * *
C
         CALL CONMAK(TMAT,IXDIM,IYDIM,TINCR)
         CALL CONLIN(0,'SOLIDS','LABELS',1,10)
         CALL CONANG(90.0)
         CALL RASPLN(0.25)
         CALL CONTUR(1,'LABELS','DRAW')
         CALL HEIGHT (0.05)
         CALL RLMESS('EXP 1C $',100,-59,-59)
         CALL DOT
         CALL GRID(1,1)
         CALL ENDPL(0)
         CALL DONEPL
         STOP
         END
```

APPENDIX D

TABULATED DATA

TABLE 1

ROTOMETER CALIBRATION

% Flow	ml/s	Std Dev (ml/s)
10	5.94	0.32
15	8.02	0.56
20	10.05	0.41
25	11.85	0.24
30	14.03	0.33
35	15.98	0.58
40	17.41	0.31
45	19.09	0.34
50	20.75	0.23
55	22.38	0.28
60	24.27	0.22
65	26.03	0.20
70	27.64	0.37
75	29.53	0.24

TABLE 2

TEST RESULTS

Crossflow velocity:	.13 m/s
Nozzle flow rate:	11.85 ml/s (25%)
Nozzle inside diameter:	7.144 mm
Nozzle discharge velocity (mean):	29.558 mm/s
Nozzle temperature (mean):	41.8 °C
Ambient temperature (mean):	24.9 °C
Froude number:	14.8
Michaelis-Menter equation:	
Coefficient A:	2.64284325
Coefficient B:	1.03698254

Plane	Y (mm)	ϕ (degrees)	$\dot{Q}$ (W)
A	7.327	46	1.20
B	21.370	58	6.392
C	39.688	70	15.349
D	62.889	80	18.383
E	87.313	86	27.628

LIST OF REFERENCES

1. Hilder, D.S., Entrainment Modeling of Buoyant Momentum Jets in Water, Mechanical Engineer's Thesis, Naval Postgraduate School, Monterey, CA, 1981.

2. Nickodem, P., Determination of Velocity Profiles in a Turbulent Jet in Crossflow Using a Laser Doppler Velocimeter, M.S. Thesis, Naval Postgraduate School, Monterey, CA, 1984.

3. Oak Ridge National Laboratory, Oak Ridge, Tennessee, Report ORNL-4685, Analysis of Round, Turbulent, Buoyant Jets Discharged to Flowing Stratified Ambients, by E.A. Hirst.

4. Incropera, F.P. and Dewitt, D.P., Fundamentals of Heat Transfer, John Wiley & Sons, 1981.

5. Beckwith, T.G., Buck, L.W., and Marangoni, R.D., Mechanical Measurements, 2nd ed., Addison-Wesley Publishing Company, 1971.

6. Merzkirch, W., Flow Visualization, Academic Press, 1974.

7. Caceci, M.S. and Cacheris, W.P., "Fitting Curves to Data," Byte: The Small Systems Journal, Vol. 9, pp. 340-362, May 1984.

8. Integrated Software Systems Corporation, DISSPLA(R) Users Manual, 1981.

9. Holman, J.P., Experimental Methods for Engineers, McGraw-Hill Book Company, 1978.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3.	Department Chairman, Code 69MX Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93943	1
4.	Dr. William G. Culbreth, Code 69Cb Department of Mechanical Engineering Naval Postgraduate School Montery, California 93943	5
5.	Ronald J. Matoushek R. D. #1 Waymart, Pennsylvania 18472	5

END

FILMED

4-85

DTIC